



TUGAS AKHIR - TF 141581

**OPTIMISASI PEMBAKARAN DI *ROTARY KILN* PT
SEMEN GRESIK PABRIK TUBAN DENGAN MODEL
JARINGAN SYARAF TIRUAN DAN METODE
*GENETIC ALGORITHM***

DICKY DWI ARIFIANSYAH
NRP. 2413 100 118

Dosen Pembimbing
Ir. Ya'umar, M.T.

DEPARTEMEN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - TF141581

***OPTIMIZATION OF FIRING AT ROTARY KILN
IN PT SEMEN GRESIK TUBAN PLANT WITH
ARTIFICIAL NEURAL NETWORK MODELLING
AND GENETIC ALGORITHM METHOD***

***DICKY DWI ARIFANSYAH
NRP. 2413 100 118***

***Supervisor
Ir. Ya'umar, M.T.***

***DEPARTMENT OF ENGINEERING PHYSICS
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2017***

PERNYATAAN BEBAS PLAGIARISME

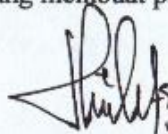
Saya yang bertanda tangan di bawah ini:

Nama : Dicky Dwi Arifiansyah
NRP : 2413100118
Departemen/ Prodi : Teknik Fisika/ S1 Teknik Fisika
Fakultas : Fakultas Teknologi Industri
Perguruan Tinggi : Institut Teknologi Sepuluh Nopember

dengan ini menyatakan bahwa Tugas Akhir saya berjudul
“OPTIMISASI PEMBAKARAN DI ROTARY KILN PT SEMEN
GRESIK PABRIK TUBAN DENGAN MODEL JARINGAN
SYARAF TIRUAN DAN METODE *GENETIC ALGORITHM*”
adalah bebas dari plagiasi. Apabila pernyataan ini terbukti tidak
benar, maka saya bersedia menerima sanksi sesuai ketentuan yang
berlaku.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Surabaya, 19 Juli 2017
Yang membuat pernyataan,

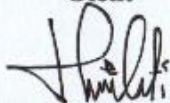


Dicky Dwi Arifiansyah
NRP. 2413100118

LEMBAR PENGESAHAN

**OPTIMISASI PEMBAKARAN DI ROTARY KILN PT
SEMEN GRESIK PABRIK TUBAN DENGAN MODEL
JARINGAN SYARAF TIRUAN DAN METODE
*GENETIC ALGORITHM***

Oleh:



Dicky Dwi Arifiansyah
NRP. 2413 100 118

Surabaya, 25 Juli 2017
Mengetahui/Menyetujui

Pembimbing



Ir. Ya'umar, M.T.
NIP. 19540406 198103 1 003

Ketua Departemen
Teknik Fisika FTI-ITS



Agus Muhammad Hatta, S.T., M.Si., Ph.D.
NIP. 19780902 200312 1 002

**OPTIMISASI PEMBAKARAN DI *ROTARY KILN* PT
SEMEN GRESIK PABRIK TUBAN DENGAN MODEL
JARINGAN SYARAF TIRUAN DAN METODE
*GENETIC ALGORITHM***

TUGAS AKHIR




Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
pada
Bidang Studi Rekayasa Instrumentasi
Program Studi S-1 Departemen Teknik Fisika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Oleh :

DICKY DWI ARIFIANSYAH

NRP. 2413 100 118

Disetujui oleh Tim Penguji Tugas Akhir :

1. Ir. Ya'umar, M.T.  (Pembimbing)
2. Hendra Cordova, S.T., M.T.  (Ketua Penguji)
3. Dr. Katherin Indriawati, S.T., M.T.  (Penguji)

SURABAYA, Juli 2017

OPTIMISASI PEMBAKARAN DI *ROTARY KILN* PT SEMEN GRESIK PABRIK TUBAN DENGAN MODEL JARINGAN SYARAF TIRUAN DAN METODE *GENETIC ALGORITHM*

Nama Mahasiswa : Dicky Dwi Arifiansyah
NRP : 2413 100 118
Departemen : Teknik Fisika FTI-ITS
Dosen Pembimbing : Ir. Ya'umar, M.T.

Abstrak

Rotary kiln pada pabrik semen mempunyai peranan penting dalam produksi semen. Tetapi dibutuhkan energi dan biaya yang besar yaitu sekitar 30% untuk mengoperasikan *rotary kiln* ini. Oleh karena itu, untuk mengurangi besarnya energi dan biaya itu diperlukan dibutuhkan sebuah metode yang tepat dan akurat dalam melakukan prediksi hasil pembakaran *rotary kiln* tersebut, sehingga dapat membantu dalam memperoleh hasil pembakaran seperti persentase FCaO dan C3S yang optimal. Untuk mengatasi masalah ini dilakukan penelitian mengenai optimisasi pembakaran di *rotary kiln* dengan model jaringan syaraf tiruan dan metode *genetic algorithm*. Nilai RMSE terbaik menggunakan metode jaringan syaraf tiruan terdapat pada *hidden node* berjumlah 6 *node* pada prediksi persentase FCaO yaitu sebesar 0,269794, dan 4 *node* pada prediksi persentase C3S yaitu sebesar 1,174429. Sedangkan, jika menggunakan metode gabungan, nilai RMSE terbaik terdapat pada *hidden node* berjumlah 2 *node* pada prediksi persentase FCaO yaitu sebesar 0.0948, dan 1 *node* pada prediksi persentase C3S yaitu sebesar 0.0878. Dapat disimpulkan dari hasil tersebut bahwa metode optimisasi *genetic algorithm* dapat menghasilkan bobot yang optimum pada jaringan syaraf tiruan, dilihat dari nilai RMSE yang dihasilkan metode gabungan lebih kecil dibandingkan dengan tanpa menggunakan metode optimisasi *genetic algorithm*.

Kata Kunci : Pembakaran, Rotary Kiln, Jaringan Syaraf Tiruan

OPTIMIZATION OF FIRING AT ROTARY KILN IN PT SEMEN GRESIK TUBAN PLANT WITH ARTIFICIAL NEURAL NETWORK MODELLING AND GENETIC ALGORITHM METHOD

Name : Dicky Dwi Arifiansyah
NRP : 2413 100 118
Department : *Department of Engineering Physics*
Supervisor : Ir. Ya'umar, M.T.

Abstract

Rotary kilns at cement plants have an important role in cement production. But it takes enormous energy and cost of about 30% to operate this rotary kiln. Therefore, to reduce the amount of energy and cost it is necessary to have a precise and accurate method of predicting the result of rotary kiln burning, so it can assist in producing combustion results such as the percentage of FCaO and C3S is optimal. To overcome this problem, research on optimization of combustion in rotary kiln with neural network model and genetic algorithm method. The best RMSE value using artificial neural network method is hidden node at 6 node at FCaO percentage prediction that is 0,269794, and 4node at prediction percentage of C3S that is equal to 1,174429. Whereas, using the combined method, the best RMSE value is found in the hidden node of 2 node in the FCaO percentage prediction of 0.0948, and 1 nodes in the C3S percentage prediction of 0.0878. It can be concluded from these results that the optimization method of genetic algorithm can produce the optimum weight and improve the accuracy of artificial neural network seen from RMSE value produced by the combined method is smaller than without using genetic algorithm optimization method.

Keywords: *Combustion, Rotary Kiln, Artificial Neural Network*

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah S.W.T, karena rahmat dan hikmat-Nya sehingga penulis diberikan kesehatan, kemudahan, dan kelancaran dalam menyusun laporan tugas akhir ini. Tidak lupa juga penulis menyampaikan ucapan terima kasih kepada keluarga dan para sahabat. Oleh karena dukungan mereka, penulis mampu menyusun laporan tugas akhir yang berjudul:

**“OPTIMISASI PEMBAKARAN DI *ROTARY KILN* PT
SEMEN GRESIK PABRIK TUBAN DENGAN MODEL
JARINGAN SYARAF TIRUAN DAN METODE *GENETIC
ALGORITHM*”**

Tugas akhir ini merupakan salah satu persyaratan akademik yang harus dipenuhi dalam Program Studi S-1 Teknik Fisika FTI-ITS. Penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Agus Muhamad Hatta, ST, MSi, Ph.D. selaku ketua jurusan Teknik Fisika ITS.
2. Ir. Ya'umar, MT selaku dosen pembimbing tugas akhir ini, yang selalu memberikan semangat dan ide-ide baru.
3. Lizda Johar Mawarani, ST, MT. selaku dosen wali penulis. Terima kasih atas bantuan, arahan, dan nasehat yang telah diberikan selama penulis menjadi mahasiswa di Jurusan Teknik Fisika FTI – ITS
4. Segenap Bapak/Ibu dosen pengajar di jurusan Teknik Fisika - ITS.
5. Para dosen dan karyawan Jurusan Teknik Fisika FTI – ITS
6. Rekan-rekan F48 dan warga Teknik Fisika - ITS, yang senantiasa memberikan motivasi dan perhatian.
7. Rekan-rekan seperjuangan TA yang telah memotivasi dan memberikan bantuan-bantuan dalam penyelesaian laporan tugas akhir ini.
8. Segenap keluarga penulis yang telah memberikan dukungan penuh terhadap penyelesaian tugas akhir ini.

Penulis menyadari bahwa mungkin masih ada kekurangan dalam laporan ini, sehingga kritik dan saran penulis terima. Semoga laporan ini dapat berguna dan bermanfaat bagi penulis dan pihak yang membacanya.

Surabaya, Juli 2017

Penulis

DAFTAR ISI

	Hal
HALAMAN JUDUL BAHASA INDONESIA	i
HALAMAN JUDUL BAHASA INGGRIS	iii
PERNYATAAN BEBAS PLAGIARISME	v
LEMBAR PENGESAHAN	vii
LEMBAR PERSETUJUAN	ix
ABSTRAK	xi
<i>ABSTRACT</i>	xiii
KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
 BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Tujuan	3
1.4 Lingkup Kerja	4
 BAB II TINJAUAN PUSTAKA	
2.1 Sistem Operasi <i>Rotary Kiln</i>	5
2.2 Pembakaran di <i>Rotary Kiln</i>	6
2.3 Jenis Batubara	8
2.4 Jaringan Syaraf Tiruan	9
2.5 Metode Pelatihan <i>Levenberg-Marquardt</i>	10
2.6 <i>Root Mean Square Error</i> (RMSE)	12
2.7 Aplikasi Jaringan Syaraf Tiruan pada Matlab	13
2.8 <i>Genetic Algorithm</i>	15
 BAB III METODE PENELITIAN	
3.1 Pengumpulan Data Input dan Output dan Operasional Rotary Kiln Di Pabrik Tuban Semen Gresik	18
3.2 Studi Literatur	18

	Hal
3.3 Pengujian Data	18
3.4 Pembuatan Program JST	19
3.5 Validasi Hasil Pemodelan	21
3.6 Optimisasi dengan <i>Genetic Algorithm</i>	21
3.7 Simulasi	26
3.8 Analisa Hasil Perhitungan	26
BAB IV HASIL DAN PEMBAHASAN	
4.1 Pengujian Data <i>Rotary Kiln</i>	27
4.2 Prediksi Hasil Pembakaran Rotary Kiln dengan menggunakan Jaringan Syaraf Tiruan	34
4.3 Prediksi Hasil Pembakaran Rotary Kiln dengan menggunakan Jaringan Syaraf Tiruan dan Optimisasi Genetic Algorithm	46
4.4 Parameter Arsitektur Jaringan Syaraf Tiruan dari Hasil Perancangan	52
4.5 Perbandingan Hasil Antara Kedua Metode	56
BAB V PENUTUP	
5.1 Kesimpulan	59
5.2 Saran	59
DAFTAR PUSTAKA	61
LAMPIRAN	65
BIOGRAFI PENULIS	

DAFTAR GAMBAR

	Hal
Gambar 2.1	Alur proses <i>rotary kiln</i> 5
Gambar 2.2	Model neuron sederhana 10
Gambar 2.3	Tampilan GUI <i>Neural Network Start</i> 14
Gambar 2.4	Contoh baris perintah program Jaringan Syaraf Tiruan pada Matlab 14
Gambar 2.5	Arsitektur JST dengan satu input layer (13 variabel input), satu hidden layer yang memiliki 20 hidden node dan 1 output layer pada Matlab. 15
Gambar 2.6	Crossover dengan pindah silang satu titik 16
Gambar 2.7	Mutasi Gen 16
Gambar 3.1	Diagram alir penelitian 17
Gambar 3.2	Arsitektur JST yang digunakan pada penelitian Tugas Akhir 19
Gambar 3.3	Diagram alir pembuatan program JST 20
Gambar 3.4	Diagram alir optimisasi genetic algorithm 22
Gambar 4.1	Hipotesis awal diagram jalur antar variabel rotary kiln 34
Gambar 4.2	Diagram jalur untuk prediksi persentase FCaO 36
Gambar 4.3	Diagram jalur untuk prediksi persentase C3S 37
Gambar 4.4	Grafik data <i>kiln feed</i> 38
Gambar 4.5	Grafik data <i>kiln coal</i> 38
Gambar 4.6	Grafik data <i>kiln speed</i> 39
Gambar 4.7	Grafik data sekam 39
Gambar 4.8	Grafik data FCaO 40
Gambar 4.9	Grafik data C3S 40
Gambar 4.10	Hasil pelatihan data persentase FCaO 42
Gambar 4.11	Hasil validasi data persentase FCaO 42
Gambar 4.12	Arsitektur pada metode JST prediksi FCaO 43
Gambar 4.13	Hasil pelatihan data persentase C3S 45
Gambar 4.14	Hasil validasi data persentase C3S 45
Gambar 4.15	Arsitektur pada metode JST prediksi C3S 46

Gambar 4.16	Arsitektur pada metode JST-GA prediksi FCaO	48
Gambar 4.17	Grafik optimisasi pada prediksi persentase FCaO	49
Gambar 4.18	Arsitektur pada metode JST-GA prediksi FCaO	51
Gambar 4.19	Grafik optimisasi pada prediksi persentase C3S	51
Gambar 4.20	Fungsi aktivasi sigmoid tangen pada JST	52

DAFTAR TABEL

	Hal
Tabel 4.1 Analisa regresi antara seluruh variabel <i>rotary kiln</i> dengan persentase FCaO	28
Tabel 4.2 Analisa regresi antara masing-masing variabel <i>rotary kiln</i> dengan persentase FCaO	29
Tabel 4.3 Analisa regresi antara seluruh variabel <i>rotary kiln</i> dengan C3S	30
Tabel 4.4 Analisa regresi antara masing-masing variabel <i>rotary kiln</i> dengan persentase C3S	31
Tabel 4.5 Koefisien korelasi dengan variabel dependen persentase FCaO	32
Tabel 4.6 Koefisien korelasi dengan variabel dependen persentase C3S	33
Tabel 4.7 Nilai RMSE prediksi persentase FCaO dengan menggunakan jaringan syaraf tiruan	41
Tabel 4.8 Nilai RMSE prediksi persentase C3S dengan menggunakan jaringan syaraf tiruan	44
Tabel 4.9 Nilai RMSE prediksi persentase FCaO dengan menggunakan jaringan syaraf tiruan dan optimisasi <i>genetic lgorithm</i>	47
Tabel 4.10 Nilai RMSE prediksi persentase C3S dengan menggunakan jaringan syaraf tiruan dan optimisasi <i>genetic algorithm</i>	50
Tabel 4.11 Faktor bobot optimum dari metode JST untuk prediksi persentase FCaO	53
Tabel 4.12 Faktor bobot optimum dari metode JST untuk prediksi persentase C3S	54
Tabel 4.13 Faktor bobot optimum dari metode JST-GA untuk prediksi persentase FCaO	55
Tabel 4.14 Faktor bobot optimum dari metode JST-GA untuk prediksi C3S	55
Tabel 4.15 Perbandingan hasil terbaik antara metode JST dan metode gabungan JST-GA	56

BAB I

PENDAHULUAN

1.1. Latar Belakang

Industri di Indonesia saat ini sedang berkembang pesat dalam rangka melaksanakan pembangunan nasional. Sejalan dengan meningkatnya laju pembangunan, konsumsi energi di Indonesia juga terus meningkat dari tahun ke tahun. Peningkatan ini terjadi hampir pada semua sektor, baik sektor industri, transportasi, komersial, rumah tangga, pembangkit listrik dan sektor lainnya (Ayu, Hadi, & Sarwono, 2016).

Berdasarkan buku *Outlook Energi Indonesia*, di tahun 2013 sektor industri merupakan sektor dengan pangsa konsumsi energi final terbesar yaitu sebesar 33%. Diikuti oleh sektor rumah tangga sebesar 27% dan sektor transportasi sebesar 27%. Sedangkan sektor komersial, sektor lain, dan penggunaan untuk bahan baku adalah 10%

Tingginya konsumsi energi di sektor industri memungkinkan terjadinya pemborosan energi yang besar pula. Sehingga diperlukan upaya untuk dapat meningkatkan efisiensi penggunaan energi pada industri. Di tahun 2013, kementerian ESDM mengelompokkan industri di Indonesia menjadi Sembilan kategori. Dari kesembilan kategori tersebut, industri semen dan barang galian bukan logam menduduki peringkat teratas sebagai industri mengkonsumsi energi terbesar (BPPT, 2013).

Industri semen adalah industri yang proses produksinya merupakan *high energy* karena membutuhkan banyak bahan bakar, baik pada *preheater* maupun *Kiln* (pembakar utama). Sekitar 30 – 40 % dari *total production cost* produksi semen hanya digunakan untuk membiayai kebutuhan bahan bakar (Madloul & al, 2011). Sehingga dapat disimpulkan bahwa adanya pemborosan dari segi biaya dan penggunaan bahan bakar, oleh karena itu proses pembakaran pada produksi semen memerlukan pengelolaan energi yang baik, agar dapat dapat diminimalisir biaya produksi dan juga gas berbahaya yang dibuang. Salah satunya dengan melakukan optimisasi.

Optimisasi adalah suatu proses untuk mencapai hasil yang ideal atau optimal (nilai efektif yang dapat dicapai). Optimisasi dapat diartikan sebagai suatu bentuk mengoptimalkan sesuatu hal yang sudah ada, ataupun merancang dan membuat sesuatu secara optimal (Politeknik Elektronika Negeri Surabaya (PENS), n.d.).

Pada penelitian sebelumnya ada yang menggunakan metode optimisasi dengan *optimal control* dan pemodelan *MPC (Model Predictive Control)* pada *rotary cement kiln* (Teja, P, & M, 2016).

Oleh karena itu dalam tugas akhir ini akan dilakukan penelitian mengenai optimisasi pembakaran di *rotary kiln* dengan metode yang berbeda yaitu optimisasi dengan *genetic algorithm* dan pemodelan jaringan syaraf tiruan.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, maka permasalahan yang akan diangkat dalam tugas akhir ini yaitu:

1. Apakah model prediktor hasil pembakaran *rotary kiln* dengan menggunakan metode Jaringan Syaraf Tiruan dapat memprediksi hasil pembakaran *rotary kiln*?
2. Apakah model prediktor hasil pembakaran *rotary kiln* dengan menggunakan metode Jaringan Syaraf Tiruan dan optimisasi *Genetic Algorithm* dapat memprediksi hasil pembakaran *rotary kiln*?

1.3. Tujuan

Tujuan dilakukan tugas akhir ini adalah sebagai berikut:

1. Membuat model prediktor hasil pembakaran *rotary kiln* dengan menggunakan metode Jaringan Syaraf Tiruan agar dapat memprediksi hasil pembakaran *rotary kiln*.
2. Membuat model prediktor hasil pembakaran *rotary kiln* dengan menggunakan metode Jaringan Syaraf Tiruan dan optimisasi *Genetic Algorithm* agar dapat memprediksi hasil pembakaran *rotary kiln*.

1.4. Lingkup Kerja

Adapun lingkup kajian yang digunakan pada tugas akhir ini adalah:

1. Pengambilan data di Semen Gresik Pabrik Tuban.
2. Memodelkan *rotary kiln* dengan menggunakan metode Jaringan Syaraf Tiruan (JST).
3. Arsitektur Jaringan Syaraf Tiruan menggunakan *Multi Layer Perceptron* (MLP), dan metode training yang digunakan adalah metode *Levenberg-Marquadt*
4. Mengoptimisasi pembakaran di *rotary kiln* dengan metode *genetic algorithm*.
5. Menggunakan *software* MATLAB 2015b
6. Laporan dan Analisa.

“Halaman ini sengaja dikosongkan”

BAB II

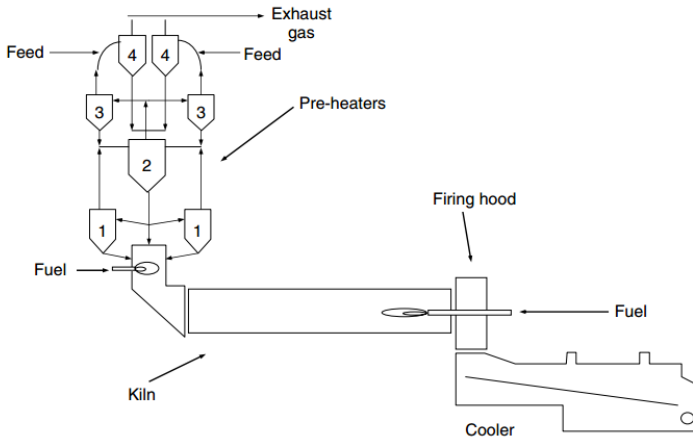
TINJAUAN PUSTAKA

2.1 Sistem Operasi *Rotary Kiln*

Rotary Kiln merupakan peralatan paling utama pada proses pembuatan semen. Fungsi utamanya adalah sebagai tempat terjadinya kontak antara gas panas dan material umpan kiln sehingga terbentuk senyawa-senyawa penyusun semen yaitu C3S, C2S, C3A dan FCaO (Boateng, 2008). Untuk mendapatkan hasil semen terbaik yaitu memiliki kadar C3S dengan rentang 55-62% dan kadar FCaO <2%

Rotary kiln di Semen Gresik pabrik Tuban mempunyai spesifikasi silinder baja dengan diameter 5,6 m dan Panjangnya 84 m, dan ditumpu oleh tiga buah *Tire*. Setiap *Tire* ditumpu oleh sepasang *Carrying Roller*. Sudut kemiringan *Rotary Kiln* adalah 4%, dan bagian dalam *Rotary Kiln* dilapisi batu tahan api.

Perputaran kiln yang berlawanan arah dengan arah jarum jam dan dengan posisi *kiln* yang miring menyebabkan terjadinya gaya dorong umpan sehingga material bisa bergerak keluar kearah *clinker cooler* setelah mengalami kontak dengan gas panas.



Gambar 2.1 Alur proses rotary kiln (Boateng, 2008)

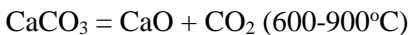
Umpan kiln dari preheater akan masuk melalui *inlet chamber*. Tenaga gerak dari *motor* dan *main gear* menyebabkan kiln berputar. Perputaran pada *kiln* diatur oleh *girth gear* yang berfungsi sebagai pengaman dan mengurangi beban *main gear*. Karena pengaruh kemiringan dan gaya putar kiln, maka umpan kiln akan bergerak perlahan disepanjang kiln. Dari arah yang berlawanan gas panas hasil pembakaran batu bara dihembuskan oleh *burner*, sehingga terjadi kontak panas dan perpindahan panas antara umpan kiln dengan gas panas. Kontak panas tersebut akan mengakibatkan terjadinya reaksi kimia untuk membentuk komponen semen. Pembakaran akan terus berlangsung sampai terbentuk klinker dan akan keluar menuju *clinker cooler*.

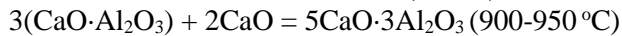
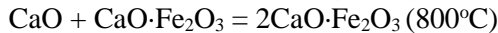
2.2 Pembakaran di *Rotary Kiln*

Di dalam rotary kiln selain jumlah panas yang dibutuhkan untuk pembakaran raw mix harus terpenuhi, perlu juga diperhatikan bentuk nyala saat pembakaran bahan bakar pada burner. Bentuk nyala ini mempengaruhi kualitas klinker yang dihasilkan. Kedua parameter ini dipengaruhi oleh proses pembakaran saat bahan bakar mulai keluar dari ujung burner hingga habis terbakar. Secara umum, pembakaran terjadi melalui 4 tahapan proses, yaitu: Pencampuran - Penyalaan - Reaksi Kimia - Penyebaran Panas / Produk Pembakaran (Boateng, 2008). Selama proses pembakaran, material akan melewati 3 zona dalam kiln dengan *range* suhu yang berbeda-beda sehingga dalam kiln akan terjadi reaksi kimia pembentukan senyawa penyusun semen. Zona-zona tersebut, yaitu: zona dekomposisi, zona transisi, dan zona sintering. Berikut ini proses kimia yang terjadi selama proses pembakaran di *rotary kiln* (Boateng, 2008):

1. Zona dekomposisi

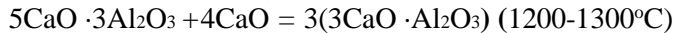
Saat memasuki daerah dekomposisi sejumlah kecil CaO , $\text{Al}_2\text{O}_3(\text{CA})$, $\text{CaO}\cdot\text{Fe}_2\text{O}_3$ (CF), $2\text{CaO}\cdot\text{Fe}_2\text{O}_3$, dan $5\text{CaO}\cdot 3\text{Al}_2\text{O}_3$ (C_5A_3) terbentuk dari reaksi dibawah ini





2. Zona transisi

Reaksi utama di zona ini adalah reaksi ekototermik yang dimulai dengan silica (C_2S), ($\Delta H = +603 \text{ kJ/kg C}_2\text{S}$) yang diikuti oleh pembentukan dari C_4AF ($\Delta H = +109 \text{ kJ/kg C}_4\text{AF}$), dan C_3A ($\Delta H = +37 \text{ kJ/kg C}_3\text{A}$). menjadi:



3. Zona sintering

Pada zona ini reaksi fase liquidus terbentuk. Komponen utamanya, C_3S ($\Delta H = +448 \text{ kJ/kg C}_3\text{S}$), dibentuk oleh reaksi antara C_2S Dibentuk sebelumnya dan setiap kapur bebas tersedia sebagai,



Untuk mendapatkan bentuk nyala yang diinginkan merupakan pekerjaan yang cukup kompleks sebab selain dengan mengatur aliran di burner tip, bentuk nyala juga dipengaruhi oleh kondisi di dalam kiln itu sendiri. Ada dua kemungkinan pengaturan bentuk nyala, yaitu (Boateng, 2008):

1. Bentuk nyala cone flame, di mana bentuk ini dihasilkan dengan komponen kecepatan aliran aksial diletakkan di bagian dalam sedang komponen radial di bagian luar.
2. Bentuk nyala hollow cone flame, di mana bentuk ini diperoleh dengan meletakkan komponen aksial di bagian luar sedang komponen radialnya di bagian dalam.

Dari bentuk nyala ada beberapa hal penting yang berpengaruh terhadap kualitas klinker yang dihasilkan, yaitu:

1. Laju Pembakaran (*burning rate*), Laju pembakaran ini sangat berpengaruh terhadap ukuran komponen *alite* (C3S) atau *Tricalcium Silicate* yang terbentuk. Komponen *alite* yang berukuran kecil akan mengakibatkan klinker yang dihasilkan tidak *dusty*, sehingga mempunyai potensi kuat tekan yang tinggi dan proses penggilingannya mudah.
2. Temperatur tertinggi (*maximum temperature*), Pada temperatur tertinggi yang sesuai akan dihasilkan klinker dengan *litre weight* yang baik, sehingga mempunyai potensi kuat tekan yang tinggi dan akan mudah digiling. Tetapi pada temperatur tertinggi yang terlalu tinggi akan dihasilkan klinker yang sifatnya berlawanan dengan sifat – sifat tersebut.
3. Waktu pembakaran (*burning time*), Kondisi ini sangat berpengaruh pada ukuran *belite* (C2S), yaitu kenaikan waktu pembakaran akan memperbesar ukuran *belite* sehingga potensi kuat tekannya akan tinggi serta akan mudah digiling. Selain itu kenaikan waktu pembakaran akan menurunkan kandungan F.CaO atau *Free Calcium* .
4. Laju pendinginan (*cooling rate*) Kondisi ini sangat berpengaruh pada warna belite yang mengindikasikan struktur kristalnya. Pendinginan yang lambat akan menghasilkan klinker dengan kuat tekan yang rendah.

2.3 Jenis Batubara

Batubara adalah bebatuan hidrokarbon yang terbentuk dari tumbuhan dalam lingkungan bebas oksigen, serta terkena pengaruh tekanan dan panas yang berlangsung sangat lama. Proses pembentukannya (*coalification*) memerlukan waktu yang sangat lama hingga jutaan tahun, mulai dari awal pembentukan yang menghasilkan gambut, lignit, subbituminus, bituminous, dan akhirnya terbentuk antrasit (Puslitbang Teknologi Mineral dan Batubara, 2011). Pada rotary kiln ada dua macam batu bara yang digunakan yaitu:

- Bituminus, mengandung 68 - 86% unsur karbon (C) dan berkadar air 8-10% dari beratnya
- Antrasit, merupakan kelas batubara tertinggi dengan unsur karbon sebesar 86%-98% dengan kadar air kurang dari 8%.

2.4 Jaringan Syaraf Tiruan

Jaringan syaraf tiruan merupakan sistem pemrosesan informasi yang terinspirasi oleh jaringan syaraf biologis. Hal tersebut membuat JST memiliki karakteristik yang hampir sama dengan jaringan syaraf biologis (Institut Teknologi Sepuluh Nopember (ITS), n.d.). Selain itu JST mampu menyelesaikan suatu permasalahan yang kompleks seperti pada *rotary kiln* ini JST dikembangkan berdasarkan asumsi bahwa:

1. Pemrosesan informasi terjadi pada elemen-elemen sederhana yang disebut neuron
2. Sinyal-sinyal informasi dikirimkan antar neuron melalui penghubung
3. Tiap penghubung memiliki bobot yang berfungsi untuk memodifikasi sinyal

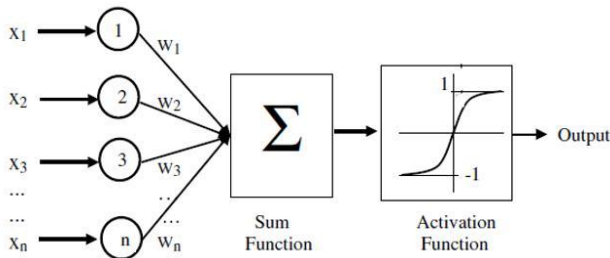
Neuron menerapkan fungsi aktivasi pada jumlah masukan yang diterima untuk menentukan nilai dari sinyal keluaran (Fausett).

Terdapat tiga hal penting yang membedakan antara JST yang satu dengan JST yang lain yaitu:

1. Arsitektur, merupakan pola hubungan antar neuron
2. *Training, learning* atau algoritma merupakan metode yang digunakan untuk menyesuaikan bobot penghubung
3. Fungsi aktivasi

JST terdiri dari sejumlah besar elemen dasar disebut neuron, unit, sel atau *node*. Keempat istilah tersebut memiliki pengertian yang sama. Neuron terdiri dari tiga elemen pembentuk yaitu (Fausett) :

1. Himpunan unit-unit yang dihubungkan oleh jalur penghubung. Jalur tersebut memiliki bobot yang berbeda satu dengan lainnya. Bobot tersebut berfungsi memodifikasi sinyal, dimana bobot positif akan memperkuat sinyal sementara bobot negatif akan memperlemah sinyal.
2. Unit penjumlahan, yang berfungsi untuk menjumlahkan sinyal-sinyal masukan yang sudah dikalikan dengan bobot
3. Fungsi aktivasi, yang berfungsi untuk menentukan apakah sinyal dari neuron masukan akan diteruskan ke neuron lain atau tidak.



Gambar 2.2 Model neuron sederhana (*Fausett*)

Gambar 2.2 di atas menunjukkan model neuron sederhana yang terdiri dari n buah bobot (w_1, w_2, w_n), fungsi aktivasi (f) dan *output*. *Input* x_1, x_2 , hingga x_n diteruskan melewati penghubung dengan bobot w yang memodifikasi nilainya menjadi $w_1x_1, w_2x_2, \dots, w_nx_n$. Keseluruhan nilai *input* yang telah dimodifikasi ini kemudian dijumlahkan sesuai dengan persamaan:

$$\text{Net} = \sum_1^n W_n \cdot x_n \dots\dots\dots (2.1)$$

Nilai ini kemudian diteruskan ke fungsi aktivasi (f) yang menghasilkan nilai keluaran jaringan sebesar:

$$\text{output} = f(\text{net}) \dots\dots\dots (2.2)$$

2.5 Metode Pelatihan Levenberg-Marquardt

Algoritma *Levenberg-Marquardt* merupakan salah satu jenis dari algoritma pelatihan *Artificial Neural Network Backpropagation* dengan dua jenis perhitungan, yaitu perhitungan maju dan perhitungan mundur. Langkah dasar algoritma *Levenberg-Marquardt* adalah penentuan matriks Hessian untuk mencari bobot-bobot dan bias koneksi yang digunakan. Matriks Hessian merupakan turunan kedua dari fungsi kinerja terhadap masing-masing komponen bobot dan bias. Untuk memudahkan proses komputasi, matriks Hessian diubah dengan pendekatan secara iteratif pada masing-masing *epoch* selama algoritma pelatihan berjalan. Proses proses perubahannya dilakukan dengan menggunakan fungsi gradien. Jika fungsi kerja yang digunakan

berbentuk kuadrat error (Atiliani, 2013). Secara singkat perhitungan *Levenberg-Marquardt* dapat dijelaskan sebagai berikut (Mokosuli, Weku, & Latumakulita):

Langkah 0:

- a. Inisialisasi bobot awal dengan bilangan acak kecil
- b. Inisialisasi *epoch* 0, $MSE \neq 0$
- c. Tetapkan maksimum *epoch* parameter *Levenberg-Marquardt* ($\eta > 0$), faktor τ dan target *error*.

Langkah 1:

Jika kondisi penghentian belum terpenuhi (*epoch* < maksimum atau $MSE > \text{target error}$), lakukan langkah berikutnya.

Langkah 2:

$Epoch = epoch + 1$

Untuk setiap pasangan data pelatihan lakukan langkah berikutnya

Langkah 3:

Unit *output* Y menerima target pola yang berhubungan dengan pola input pelatihan. Jika diberikan N pasangan input data pelatihan $(\mathbf{x}_r, \mathbf{t}_r), r = 1, 2, \dots, N$ (2.3)

Dengan:

\mathbf{x}_r = Input

\mathbf{t}_r = Target yang akan dicapai.

Kesalahan pada suatu data pelatihan ke-r didefinisikan sebagai:

$\mathbf{e}_r = \mathbf{t}_r - \mathbf{y}_r$ (2.4)

Dengan:

\mathbf{e}_r = Kesalahan pada unit output

\mathbf{t}_r = Keluaran yang diinginkan atau target

\mathbf{y}_r = Keluaran aktual

\mathbf{e} adalah vektor kesalahan berukuran $N \times 1$ yang tersusun dari $\mathbf{e}_r, r = 1, 2, \dots, N$. \mathbf{e} dapat dituliskan sebagai berikut:

$\mathbf{e} = [\mathbf{e}_1 \ \mathbf{e}_2 \ \dots \ \mathbf{e}_N]^T$ (2.5)

Misal bobot dari bias koneksi dinyatakan dalam vektor \mathbf{w} , \mathbf{w} merupakan vektor berukuran $((2 + 1)p + 1) \times 1$ dapat dituliskan sebagai:

$$\mathbf{w} = [\mathbf{w}_j \quad \mathbf{b}_2 \quad \mathbf{v}_{ij} \quad \mathbf{b}_1] \quad (2.6)$$

Kesalahan suatu pelatihan jaringan oleh vektor bobot dan bias koneksi \mathbf{w} pada suatu data pelatihan ke- r menjadi:

$$\mathbf{e}_r(\mathbf{w}) = (\mathbf{t}_r - \mathbf{y}_r) = (\mathbf{t}_r - \mathbf{f}\mathbf{x}_r, \mathbf{w}) \quad (2.7)$$

Vektor kesalahan oleh vektor bobot dan bias koneksi \mathbf{w} menjadi $\mathbf{e}(\mathbf{w})$ berukuran $N \times 1$ yang tersusun dari $\mathbf{e}_r(\mathbf{w})$ dengan $r = 1, 2, \dots, N$. Hitung fungsi jumlah kuadrat *error* dengan persamaan

$$\mathbf{E}(\mathbf{w}) = \frac{1}{2} \mathbf{e}^T(\mathbf{w}) \mathbf{e}(\mathbf{w}) \quad (2.8)$$

Hitung matriks Jacobian untuk vektor bobot dan bias koneksi:

$$\mathbf{J}(\mathbf{w}) = \left[\frac{\partial \mathbf{e}_r}{\partial \mathbf{w}} \right]_{N \times ((2+n)p+1)} \quad (2.9)$$

Untuk $r = 1, 2, 3, \dots, N$

- a. Hitung matriks Hessian untuk vektor bobot dan bias dengan persamaan:

$$\mathbf{H}(\mathbf{w}) = [\mathbf{J}^T(\mathbf{w}) \mathbf{J}(\mathbf{w}) + \eta \mathbf{I}] \quad ((2+n)p+1) \times ((2+n)p+1) \quad (2.10)$$

- b. Hitung perubahan vektor bobot dan bias dengan persamaan:

$$\Delta \mathbf{w} = -[\mathbf{H}(\mathbf{w})]^{-1} \mathbf{J}^T(\mathbf{w}) \mathbf{e}(\mathbf{w}) \quad ((2+n)p+1) \times ((2+n)p+1) \quad (2.11)$$

- c. Hitung vektor bobot dan bias baru

$$\mathbf{w}(\text{baru}) = \mathbf{w}(\text{lama}) + \Delta \mathbf{w} \quad (2.12)$$

- d. Hitung kesalahan yang terjadi oleh bobot dan bias koneksi yang baru

$$\mathbf{E}(\mathbf{w}(\text{baru})) = \frac{1}{2} \mathbf{e}(\mathbf{w}(\text{baru}))^T \mathbf{e}(\mathbf{w}(\text{baru})) \quad (2.13)$$

- e. Bandingkan $\mathbf{E}(\mathbf{w})$ dengan $\mathbf{E}(\mathbf{w}(\text{baru}))$

- Jika $\mathbf{E}(\mathbf{w}) \leq \mathbf{E}(\mathbf{w}(\text{baru}))$ maka didapatkan $\eta = \eta * \tau$ dan kembali ke langkah a.
- Jika $\mathbf{E}(\mathbf{w}) > \mathbf{E}(\mathbf{w}(\text{baru}))$ maka didapatkan η / τ $\mathbf{w}(t+1) = \mathbf{w}(t) + \Delta \mathbf{w}$ kembali ke langkah

2.6 Root Mean Squared Squared Error (RMSE)

Root Mean Square Error (RMSE) biasanya digunakan untuk mengetahui perbedaan antara nilai yang di prediksi oleh model dengan nilai aktual yang telah di observasi. RMSE menghitung

rata-rata error berdasarkan pada nilai kuadrat error, dan tidak mengindikasikan arah nilai deviasi. RMSE lebih baik jika digunakan pada nilai error yang besar dibandingkan nilai error yang kecil, yang mana akan lebih baik jika error besar tersebut tidak diinginkan (Litta A. J, 2012). RMSE juga lebih baik digunakan dalam peramalan konservatif. Berikut ini adalah rumus dari RMSE

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (F_i - O_i)^2} \dots\dots\dots (2.14)$$

Dimana:

F_i = Nilai prediksi

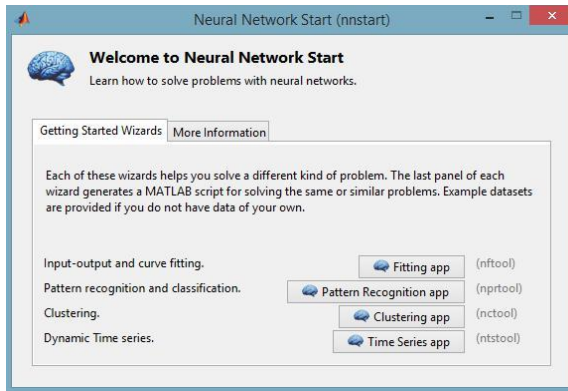
O_i = Nilai Aktual

N = Jumlah Data

2.7 Aplikasi Jaringan Syaraf Tiruan pada Matlab

Aplikasi Jaringan Syaraf Tiruan merupakan salah satu produk yang tersedia secara opsional pada Matlab. Aplikasi JST pada Matlab dikenal dengan nama *Neural Network Toolbox* (NNT). *Toolbox* ini menyediakan berbagai fungsi dan aplikasi untuk melakukan pemodelan sistem *nonlinear* yang sangat kompleks dengan menerapkan prinsip-prinsip JST. NNT mendukung pelatihan dengan supervisi dan tanpa supervisi. Dengan bantuan NNT, pengguna dapat merancang berbagai jenis Jaringan Syaraf Tiruan, melatih, dan mensimulasikan JST tersebut. NNT dapat digunakan untuk berbagai aplikasi seperti fitting data, pengenalan pola, pengelompokan data, prediksi, dan pemodelan sistem dinamik (Demuth, Beale, & Hagan, 2004).

Terdapat beberapa cara untuk merancang JST menggunakan NNT yang ada pada Matlab. Pertama yaitu dengan menggunakan *toolbox* berbasis *Graphical User Interface* (GUI) seperti tampak pada Gambar 2.5 di bawah. *Toolbox* ini dapat ditampilkan dengan mengetikkan perintah “nnstart” pada *command line*.



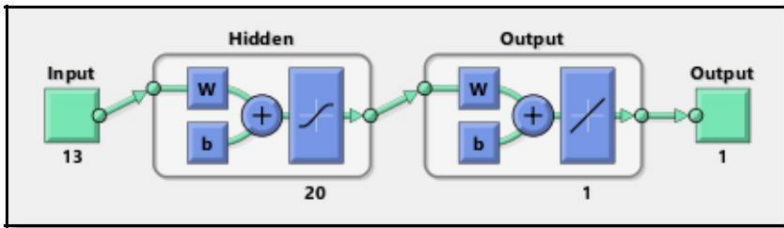
Gambar 2.3 Tampilan GUI Neural Network Start (*Demuth, Beale, & Hagan, 2004*)

Cara kedua yaitu dengan menuliskan *script* yang berisi fungsi-fungsi JST yang sudah disediakan oleh *Neural Network Toolbox* Matlab seperti yang ditunjukkan pada Gambar 2.6

```
1 - load house_dataset
2 - net = feedforwardnet(20);
3 - [net,tr] = train(net,houseInputs,houseTargets);
```

Gambar 2.4 Contoh baris perintah program Jaringan Syaraf Tiruan pada Matlab (*Demuth, Beale, & Hagan, 2004*)

Script di atas hanya terdiri dari tiga baris, namun sudah mencakup langkah-langkah perancangan JST mulai dari inisialisasi hingga pelatihan JST. Gambar 2.5 di bawah menunjukkan arsitektur JST yang dibuat berdasarkan perintah pada Gambar 2.4 JST dengan arsitektur demikian umumnya disebut memiliki konfigurasi 13 – 2 – 1.



Gambar 2.5 Arsitektur JST dengan satu input layer (13 variabel input), satu hidden layer yang memiliki 20 hidden node dan 1 output layer pada Matlab. (Demuth, Beale, & Hagan, 2004)

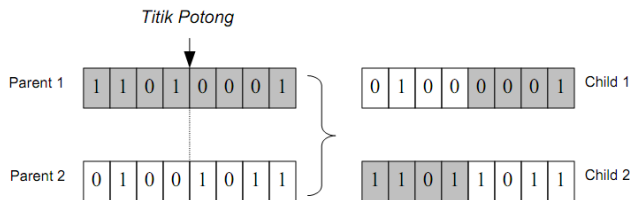
2.8 Genetic Algorithm

Genetic algorithm dikembangkan oleh John Holland pada tahun 1970. *Genetic algorithm* adalah algoritma optimisasi yang meniru prinsip genetika alam dan seleksi alam. *Genetic algorithm* melakukan pencarian dengan arah yang acak menggunakan alternatif yang telah diberikan untuk menemukan alternatif terbaik dengan menyesuaikan pada kriteria *fitness*. *Fitness* adalah suatu nilai yang akan dimaksimalkan atau diminimalkan. Sebuah populasi awal dari kromosom diambil untuk menghasilkan keturunan yang dapat bertahan hidup untuk membuat generasi berikutnya (Gill, Singh, & Singh, 2010). Operasi dasar dari *Genetic algorithm* adalah *selection*, *crossover*, dan *mutation*. Hal-hal yang harus dilakukan dalam algoritma genetika adalah (Politeknik Elektronika Negeri Surabaya (PENS), n.d.):

1. Mendefinisikan individu, dimana individu menyatakan salah satu solusi (penyelesaian) yang mungkin dari penyelesaian yang diangkat
2. Mendefinisikan nilai fitness, yang merupakan ukuran baik tidaknya sebuah individu atau baik tidaknya solusi yang didapatkan.
3. Menentukan proses pembangkitan populasi awal, hal ini biasanya dilakukan dengan menggunakan pembangkitan acak seperti *random-walk*.

4. Menentukan proses seleksi yang akan digunakan, dengan menggunakan mesin roulette, yang merupakan metode yang paling sederhana dan sering dikenal dengan nama stochastic sampling with replacemenet
5. Menentukan proses perkawinan silang (cross-over) dan mutasi den yang akan digunakan
 - a. Perkawinan silang (crossover), adalah operator dari *genetic algorithm* yang melibatkan dua induk untuk membentuk kromosom baru. Perkawinan silang menghasilkan titik baru dalam ruang pencarian yang siap untuk diuji.

- Pindah Silang Satu Titik



Gambar 2.6 Crossover dengan pindah silang satu titik

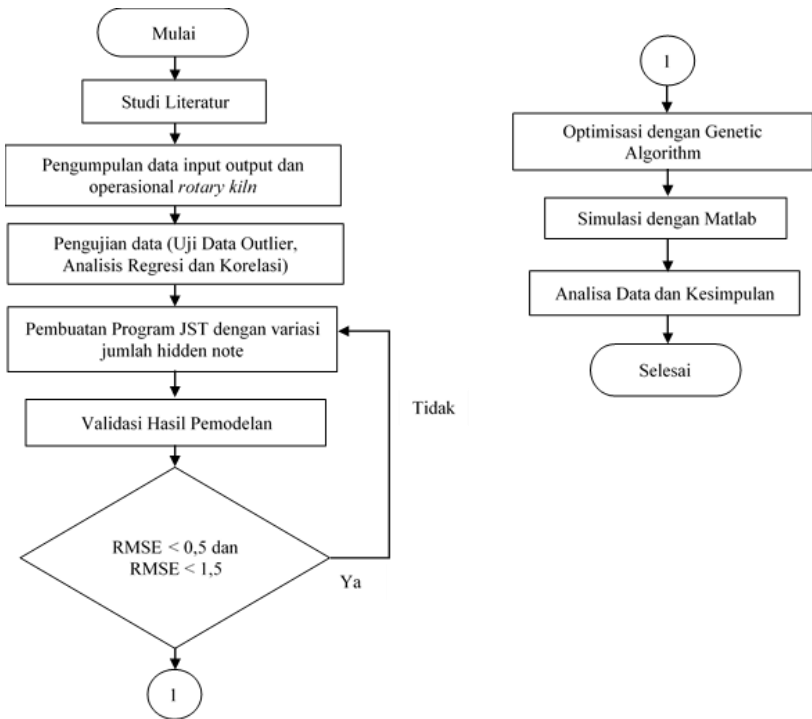
- b. Mutasi gen, operator ini berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi.



Gambar 2.7 Mutasi Gen

BAB III METODE PENELITIAN

Diagram alir dari penelitian ini ditunjukkan pada Gambar (3.1).



Ket:

RMSE < 0,5 : FCaO

RMSE < 1,5 : C3S

Gambar 3. 1 Diagram alir penelitian

3.1 Pengumpulan Data Input dan Output dan Operasional Rotary Kiln Di Pabrik Tuban Semen Gresik

Pengumpulan data yang diperlukan dalam tugas akhir ini yakni berupa data spesifikasi, desain dan operasional *rotary kiln*.

3.2 Studi Literatur

Studi literatur merupakan kegiatan yang dilakukan untuk mempelajari prinsip dasar untuk melakukan penelitian, yaitu variabel prediksi hasil pembakaran *rotary kiln*, Jaringan Saraf Tiruan, Algoritma Genetik, Literatur yang digunakan adalah buku, jurnal penelitian ilmiah, artikel ilmiah, laporan tugas akhir dan juga informasi-informasi yang diperoleh dari pihak PT. Semen Gresik Pabrik Tuban.

3.3 Pengujian Data

Pengujian data bertujuan untuk memastikan bahwa adanya hubungan antara variabel-variabel input terhadap variabel output. Pengujian data dan analisis data menggunakan software IBM SPSS Statistics 24. Uji data dan analisis data yang dilakukan, meliputi:

a. Uji data *outlier*

Data *outlier* adalah data yang secara nyata berbeda dengan data-data yang lain. Untuk mendeteksi apakah ada data outlier atau tidak, dilihat dari nilai standarisasi data (z). Jika sebuah data outlier, maka nilai z yang didapat lebih besar dari angkat $+2.5$ atau lebih kecil dari angkat -2.5

b. Analisis Regresi

Analisis regresi digunakan untuk memprediksi seberapa jauh perubahan nilai variabel dependen, bila nilai variabel independen di manipulasi atau dirubah-rubah atau dinaik-turunkan. Bila koefisien korelasi (r) tinggi, pada umumnya koefisien regresi (b) juga tinggi, sehingga daya prediktifnya akan tinggi

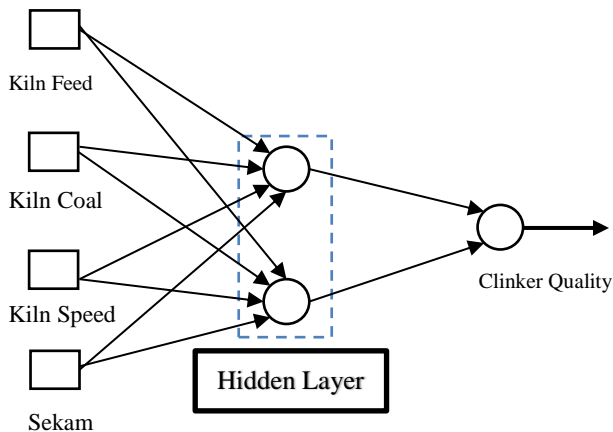
c. Analisis Korelasi

Variabel *rotary kiln* yang dimiliki yaitu *kiln feed*, *kiln coal*, *kiln speed*, sekam, FCaO, C3S dan temperatur akan

dianalisis korelasinya untuk mengetahui korelasi antar variabel *input* terhadap variabel *output* maka dilihat dari nilai koefisien korelasi. Koefisien korelasi positif terbesar adalah 1 dan koefisien korelasi negatif terbesar adalah -1, sedangkan yang terkecil adalah 0.

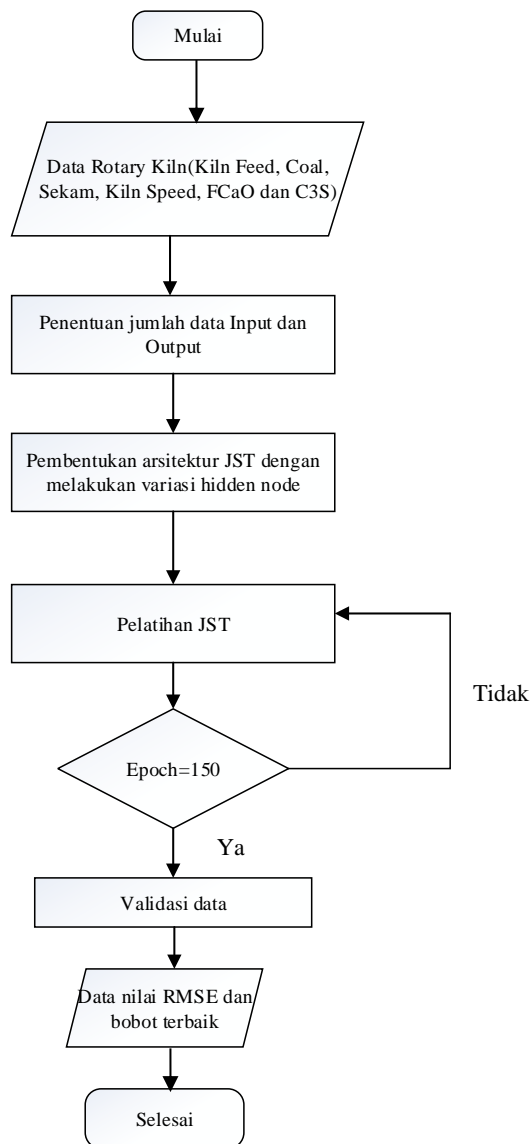
3.4 Pembuatan Program JST

Pada penelitian ini, arsitektur JST yang digunakan adalah JST *multilayer* dengan algoritma pelatihan *Levenberg-Marquardt*. Arsitektur JST tersebut ditunjukkan pada Gambar 3.2 di bawah.



Gambar 3. 2 Arsitektur JST yang digunakan pada penelitian Tugas Akhir

Keempat kotak menunjukkan elemen bias yang bertindak sebagai variabel input pada jaringan syaraf tiruan. Sementara itu, garis panah yang menghubungkan masing-masing node pada arsitektur di atas memiliki bobot yang bernilai acak. Nilai bobot awal untuk masing-masing pelatihan akan berbeda satu sama lain sehingga bobot akhir masing-masing pelatihan pun akan berbeda.



Gambar 3. 3 Diagram alir pembuatan program JST

Algoritma pertama yang digunakan untuk melakukan prediksi hasil pembakaran *rotary kiln* adalah algoritma yang hanya menggunakan model jaringan syaraf tiruan. Dapat dilihat pada diagram alir, langkah pertama yaitu mengambil data input dari *rotary kiln* sesuai dengan variabel hasil pembakaran yang dibutuhkan dalam melakukan prediksi jumlah *clinker*, persentase FCaO dan C3S. Data tersebut kemudian dilakukan normalisasi sehingga data dari seluruh variabel cuaca memiliki *range* yang sama yaitu 0 hingga 1. Setelah itu, jumlah data untuk pelatihan dan validasi ditentukan dan divariasikan. Terdapat variasi pada arsitekturnya yaitu variasi jumlah *hidden node*. Jumlah *hidden node* yang digunakan yaitu dimulai dari 1 buah *hidden node* hingga 10 buah *hidden node*. Kemudian dilakukan pelatihan data menggunakan algoritma pembelajaran dari jaringan syaraf tiruan hingga *epoch* maksimal yaitu 300 *epoch*. Hasil dari data pelatihan tersebut kemudian di validasi dan dilihat nilai RMSE yang terbaik dari semua variasi yang telah dilakukan. Sehingga akan diperoleh jumlah data validasi dan jumlah *hidden node* yang terbaik untuk melakukan prediksi hasil pembakaran di *rotary kiln*.

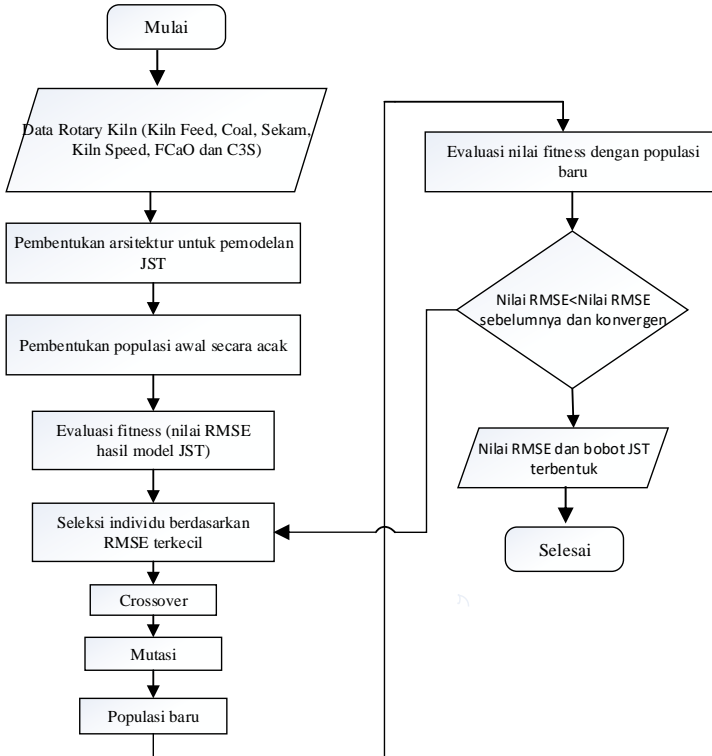
3.5 Validasi Hasil Pemodelan

Dasar penentuan model JST terbaik dari 10 jenis JST dengan 10 *hidden node* berbeda terletak pada nilai RMSE (*Mean Squared Error*) rata-rata validasi masing-masing JST tersebut.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (F_i - O_i)^2} \dots\dots\dots (3.1)$$

3.6 Optimisasi dengan *Genetic Algorithm*

Optimisasi yang digunakan adalah *Genetic Algorithm*. *Genetic Algorithm* merupakan metode *adaptive* yang biasa digunakan untuk memecahkan suatu pencarian nilai dari sebuah masalah optimisasi. Berikut ini adalah diagram alir dari *genetic algorithm*



Gambar 3. 4 Diagram alir optimisasi genetic algorithm

Berikut ini adalah langkah-langkah metode gabungan dari jaringan syaraf tiruan dan *genetic algorithm*

Langkah 1:

Mendefinisikan individu, individu adalah sekumpulan dari bobot JST.

Individu yang digunakan sebanyak 20

Iterasi yang digunakan sebanyak 10

Langkah 2:

Membangkitkan populasi awal secara acak.

Langkah 3:

Evaluasi nilai *fitness*

Nilai *fitness* adalah $\frac{1}{RMSE}$, dengan nilai RMSE sebagai berikut:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (F_i - O_i)^2} \dots\dots\dots (3.2)$$

dimana:

F_i = Nilai prediksi

O_i = Nilai Aktual

N = Jumlah Data

Nilai prediksi merupakan nilai dari keluaran JST, untuk mendapatkan nilai prediksi sebagai berikut:

- Pertama, menghitung nilai sinyal masukan ke *node* tersembunyi

$$y_{in,j} = w_{0j} + \sum_{i=1}^n x_i w_{ij} \dots\dots\dots (3.3)$$

dimana:

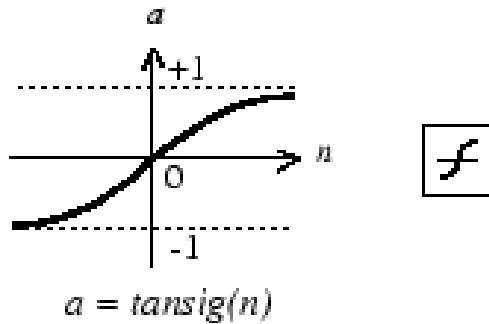
$y_{in,j}$ = sinyal masukan ke *node* tersembunyi ($j=1,2,3,\dots,n$)

w_{0j} = nilai bobot dari bias menuju *node* tersembunyi

x_i = sinyal keluaran dari *node* masukan ($i=1,2,3,\dots,m$)

w_{ij} = nilai bobot dari *node* masukan menuju *node* tersembunyi

- Kedua, menghitung sinyal keluaran dari *node* tersembunyi yang didapatkan dengan mengaplikasikan sinyal masukan pada fungsi aktivasi. Fungsi aktivasi yang dapat digunakan yaitu Tangen Sigmoid.



Gambar diatas menunjukkan fungsi aktivasi dari Tangen Sigmoid atau Tangen Hiperbolik. Pemilihan fungsi aktivasi Tangen Sigmoid untuk prediksi hasil pembakaran *rotary kiln* dikarenakan Tangen Sigmoid merupakan fungsi aktivasi yang banyak digunakan dalam kasus prediksi. Alasan banyak digunakannya fungsi aktivasi Tangen Sigmoid dikarenakan fungsi tersebut dapat memenuhi properti diantara turunan dan dirinya sendiri sehingga mudah dilakukan secara komputasi. Tangen Sigmoid juga dapat digunakan untuk pemodelan yang non linear. Tangen sigmoid memiliki interval dari -1 hingga 1. Sehingga, dapat digunakan persamaan berikut untuk mencari sinyal keluaran *node* tersembunyi:

$$y_j = \tanh(y_{in,j}) \dots\dots\dots (3.4)$$

- Ketiga, menghitung nilai sinyal masukan ke *node* keluaran
 $z_{in,k} = v_{0k} + \sum_{j=1}^n y_j v_{jk} \dots\dots\dots (3.5)$

dimana:

$z_{in,k}$ = sinyal masukan ke node keluaran ($k=1,2,3,\dots,p$)

v_{0k} = nilai bobot dari bias menuju node keluaran

y_j = sinyal keluaran dari node tersembunyi ($j=1,2,3,\dots,n$)

v_{jk} = nilai bobot dari node tersembunyi menuju node keluaran

- Terakhir, memperoleh sinyal *node* keluaran dapat diperoleh dengan mengaplikasikan sinyal masukan pada fungsi aktivasi. Fungsi aktivasi yang dapat digunakan yaitu Tangen Sigmoid. Sehingga, dapat digunakan persamaan berikut untuk mencari sinyal keluaran dari *node* keluaran adalah:

$$z_k = \tanh(z_{in.k}) \dots\dots\dots (3.6)$$

dimana:

z_k = sinyal keluaran dari node keluaran ($k=1,2,3,\dots,p$).

Nilai sinyal keluaran dari *node* keluaran ini yang disebut nilai prediksi dari keluaran JST (F_i)

Langkah 4:

Seleksi individu, pemilihan individu berdasarkan nilai *fitness*-nya. Semakin tinggi nilai *fitness* suatu individu semakin besar kemungkinan untuk terpilihnya. Teknik seleksi yang digunakan adalah *elitism*

Langkah 5:

Crossover, salah satu operator yang melibatkan dua induk untuk menghasilkan keturunan yang baru. Melakukan pertukaran gen dari dua induk secara acak.

Langkah 6:

Mutasi gen, operator yang menukar nilai gen dengan nilai inversinya, misalnya nilai gen 0 ditukar menjadi nilai gen 1. Setiap individu mengalami mutasi gen dengan probabilitas mutasi yang telah ditentukan yaitu antara 0-1. Mutasi gen dilakukan dengan memberikan nilai inversi atau menggeser nilai gen pada gen yang terpilih untuk dimutasikan.

Langkah 7:

Bila kriteria yang diinginkan belum terpenuhi maka akan dibentuk lagi generasi baru dengan mengulang langkah 3 hingga langkah 7.

Pada penelitian ini yang dioptimisasi dari pemodelan jaringan syaraf tiruan yang didapat adalah bobot nya agar ketika bobot sudah dioptimisasi maka akan mendapatkan RMSE yang minimum antara data aktual dan data hasil pemodelan, dengan fungsi obyektif nya:

$$f = 1/x$$

Dimana x adalah nilai RMSE nya, semakin kecil nilai RMSE maka semakin besar fungsi obyektifnya.

3.7 Simulasi

Hasil perhitungan optimisasi tersebut kemudian di simulasi dengan bantuan software matlab dan nantinya akan diketahui apakah sudah optimum apa belum.

3.8 Analisa Hasil Perhitungan

Dari hasil pengujian yang telah dilakukan didapatkan data-data yang kemudian di analisa untuk menjawab tujuan dalam bentuk kesimpulan tugas akhir. Setelah melakukan analisa data maka akan akan menghasilkan hasil optimasi dari pembakaran di *rotary kiln*.

BAB IV

HASIL DAN PEMBAHASAN

Dalam penelitian ini metode penentuan bobot yang digunakan pada training jaringan syaraf tiruan yaitu metode Lavenberg-Marquardt dengan struktur jaringan syaraf tiruan menggunakan Multi Layer Perceptron (MLP). Struktur input jaringan syaraf tiruan yang terdiri dari 4 *input* yaitu *kiln feed*, *kiln Coal*, *kiln speed*, sekam, 2 *hidden neuron* , dan 2 *output* yaitu C₃S, dan FCaO. Sehingga terdapat matrik 4x2 yang merupakan bobot pengali antara *input layer* ke *hidden layer*, dan 2x3 yang merupakan bobot pengali antara *hidden layer* dan *output layer*. Proses training pada dasarnya adalah untuk menentukan bobot antara *input layer* dan *hidden layer* serta bobot antara *hidden layer* dan *output layer*.

Kriteria pemodelan proses yang dihasilkan dinyatakan dalam RMSE (*Root Mean Square Error*) dimana semakin kecil nilai RMSE yang dihasilkan (mendekati nilai 0) akan menghasilkan prediksi *output* yang lebih baik.

6.1. Pengujian Data Rotary Kiln

Seperti yang sudah dijelaskan pada bab sebelumnya, uji dan analisa dilakukan dengan menggunakan *software IBM SPSS Statistics* 24. Berikut merupakan hasil dari analisa data yang digunakan untuk memilih variabel masukan untuk melakukan prediksi persentase FCaO dan C3S.

4.1.1 Analisa Regresi Antar Variabel

Analisa regresi digunakan untuk memprediksikan seberapa jauh perubahan nilai variabel dependen (terikat), bila nilai variabel independen (bebas) dimanipulasi atau diubah (Sugiyono, 2017). Hal ini nantinya akan digunakan untuk menentukan variabel *rotary kiln* apa saja yang dapat dijadikan sebagai data masukan dalam melakukan prediksi variabel hasil pembakaran *rotary kiln* yang diinginkan, khususnya variabel persentase FCaO dan C3S.

- Persentase FCaO

Variabel dependen pertama yang digunakan merupakan keluaran hasil prediksi pertama yaitu Persentase FCaO (FCaO). Variabel independen yang digunakan adalah sebuah variabel cuaca yang telah diperoleh datanya, yaitu Kiln Feed, Kiln Coal, Sekam, Kiln Speed, C3S dan Temp. Analisa regresi yang dilakukan adalah analisa regresi linear berganda, yang mana berguna untuk menganalisa hubungan antara banyak variabel independen dengan satu variabel dependen (Priyatno, 2009). Hasil analisa regresi dapat dilihat pada tabel 4.1 dan 4.2.

Tabel 4.1. Analisa regresi antara seluruh variabel *rotary kiln* dengan persentase FCaO

<i>Dependent Variable</i>	R	R²	<i>Std. Error of the Estimate</i>
FCaO	0.499	0.249	0.364

Pada tabel 4.1. nilai R menunjukkan koefisien korelasi berganda, yaitu antara variabel independen dengan dependen. Jika nilai R mendekati 1, maka hubungannya semakin tinggi. Dengan nilai R sebesar 0,499, dapat disimpulkan bahwa seluruh variabel independen memiliki hubungan yang erat dengan kecepatan angin.

Nilai R² menunjukkan koefisien determinasi dengan diubah ke bentuk persen, sehingga menunjukkan persentasi pengaruh variabel independen terhadap variabel dependen. Dengan nilai R² sebesar 0,249, artinya persentasi pengaruh seluruh variabel independen terhadap persentase FCaO adalah sebesar 24.9%, sedangkan sisanya sebesar 75,1% dipengaruhi oleh variabel lain yang tidak dimasukkan.

Standard Error of the Estimate menunjukkan ukuran kesalahan prediksi. Dalam kasus ini nilainya sebesar 0.364 yang berarti kesalahan yang dapat terjadi dalam melakukan prediksi persentase FCaO adalah sebesar 0,364 %.

Tabel 4.2. Analisa regresi antara masing-masing variabel *rotary kiln* dengan persentase FCaO

<i>Independent Variable</i>	<i>Unstandardized Coefficients</i>		<i>Standardized Coefficients</i>	<i>t</i>	<i>Sig.</i>
	<i>B</i>	<i>Std. Error</i>	<i>Beta</i>		
(<i>Constant</i>)	4,886	0,437	-	11,173	0,000
Feed(X1)	0,004	0,001	0,382	5,64	0,000
Coal(X2)	0,016	0,005	0,107	3,298	0,001
Speed(X3)	-0,002	0,000	-0,448	-6,512	0,000
Sekam(X4)	-0,03	0,003	-0,294	-9,698	0,000
Temp(X5)	0,001	0,000	0,058	1,988	0,047
C3S(X6)	-0,066	0,007	-0,295	-9,912	0,000
*Dependent Variable: FCaO (Y')					

Pada tabel 4.2. terdapat koefisien B yang terdiri dari nilai konstan (*constant*) sebesar 4,886 yang menunjukkan nilai variabel dependen jika seluruh variabel independen bernilai nol, dan juga terdapat koefisien regresi yang terletak dibawah nilai konstan pada tabel. Nilai-nilai ini kemudian dimasukkan ke dalam persamaan linear berganda, sebagai berikut.

$$Y' = 4.886 + 0.004X1 + 0.016X2 - 0.002X3 - 0.03X4 + 0.001X5 - 0.066X6 \quad (4.1)$$

Nilai beta merupakan koefisien yang menunjukkan besarnya pengaruh masing-masing variabel independen terhadap variabel persentase FCaO. Semakin nilai beta mendekati nol, maka pengaruhnya terhadap persentase semakin kecil. Dilihat dari nilai yang tertera pada tabel, variabel yang memiliki pengaruh yang besar terhadap persentase dengan nilai koefisien beta diatas 0,1 adalah Feed, Coal, Speed, Sekam dan C3S.

- Persentase C3S

Variabel dependen kedua yang digunakan merupakan keluaran hasil prediksi kedua yaitu C3S. Variabel independen yang digunakan adalah sebuah variabel cuaca yang telah diperoleh datanya, yaitu Kiln Feed, Kiln Coal, Sekam, Kiln Speed, FCaO dan Temp. Analisa regresi yang dilakukan adalah analisa regresi linear berganda, yang mana berguna untuk menganalisa hubungan antara banyak variabel independen dengan satu variabel dependen (Priyatno, 2009). Hasil analisa regresi dapat dilihat pada tabel 4.3 dan 4.4.

Tabel 4.3. Analisa regresi antara seluruh variabel *rotary kiln* dengan C3S

<i>Dependent Variable</i>	R	R²	<i>Std. Error of the Estimate</i>
C3S	0,375	0.141	1,7511

Pada tabel 4.3. nilai R menunjukkan koefisien korelasi berganda, yaitu antara variabel independen dengan variabel dependen. Jika nilai R mendekati 1, maka hubungannya semakin tinggi. Dengan nilai R sebesar 0.375, dapat disimpulkan bahwa seluruh variabel independen memiliki hubungan yang cukup erat dengan persentase C3S.

Nilai R² menunjukkan koefisien determinasi dengan diubah ke bentuk persen, sehingga menunjukkan persentasi pengaruh variabel independen terhadap variabel dependen. Dengan nilai R² sebesar 0.141, artinya persentasi pengaruh seluruh variabel independen terhadap visibilitas adalah sebesar 14,1%, sedangkan sisanya sebesar 85,9% dipengaruhi oleh variabel lain yang tidak dimasukkan.

Standard Error of the Estimate menunjukkan ukuran kesalahan prediksi. Dalam kasus ini nilainya sebesar 1,7511 yang berarti kesalahan yang dapat terjadi dalam melakukan prediksi visibilitas adalah sebesar 1,7511 %.

Tabel 4.4. Analisa regresi antara masing-masing variabel *rotary kiln* dengan persentase C3S

<i>Independent Variable</i>	<i>Unstandardized Coefficients</i>		<i>Standardized Coefficients</i>	<i>t</i>	<i>Sig.</i>
	<i>B</i>	<i>Std. Error</i>	<i>Beta</i>		
<i>(Constant)</i>	62,003	0,841	-	73,737	0,000
Feed(X1)	-0,008	0,004	-0,157	-2,138	0,033
Coal(X2)	0,116	0,023	0,176	5,129	0,000
Speed(X3)	0,000	0,002	0,015	0,199	0,832
Sekam(X4)	-0,062	0,015	-0,135	-4,004	0,000
Temp(X5)	0,000	0,002	-0,006	-0,179	0,858
FCaO(X6)	-1,519	0,153	-0,338	-9,912	0,000
*Dependent Variable: C3S (Y')					

Pada tabel 4.4. terdapat koefisien B yang terdiri dari nilai konstan (*constant*) sebesar 62,003 yang menunjukkan nilai variabel dependen jika seluruh variabel independen bernilai nol, dan juga terdapat koefisien regresi yang terletak dibawah nilai konstan pada tabel. Nilai-nilai ini kemudian dimasukkan ke dalam persamaan linear berganda, sebagai berikut.

$$Y' = 62.003 - 0.008X_1 + 0.116X_2 + 0.000X_3 - 0.062X_4 + 0.000X_5 - 1.519X_6 \quad (4.2)$$

Nilai beta merupakan koefisien yang menunjukkan besarnya pengaruh masing-masing variabel independen terhadap variabel persentase C3S. Semakin nilai beta mendekati nol, maka pengaruhnya terhadap visibilitas semakin kecil. Dilihat dari nilai yang tertera pada tabel, variabel yang memiliki pengaruh yang besar terhadap persentase C3S dengan nilai koefisien beta diatas 0,1 adalah Feed, Coal, Sekam, dan FCaO.

4.1.2 Analisa Korelasi Antar Variabel *Rotary Kiln*

Analisa korelasi dilakukan untuk melihat hubungan antara satu variabel *rotary kiln* dengan variabel *rotary kiln* lainnya yang dapat dilihat dari nilai koefisien korelasi. Dalam hal ini, analisa korelasi digunakan untuk melihat hubungan antar variabel independen sehingga dapat diketahui variabel independen mana saja yang dapat digunakan secara bersama-sama sebagai data masukan untuk melakukan prediksi. Interval koefisien korelasi antara 0 – 0,199 tingkat korelasinya sangat rendah, 0,2 – 0,399 tingkat korelasinya rendah, 0,4 – 0,599 tingkat korelasinya sedang, 0,6 – 0,799 tingkat korelasinya kuat, dan 0,8 – 1 tingkat korelasinya sangat kuat (Sugiyono, 2017).

- Persentase FCaO

Terdapat 5 variabel yang memiliki pengaruh besar terhadap perubahan persentase FCaO dilihat dari hasil analisa regresi, yaitu Feed dengan koefisien beta 0,382, Coal dengan koefisien beta 0,107, Speed dengan koefisien beta -0,448, Sekam dengan koefisien beta -0,294, dan C3S dengan koefisien beta -0,295. Kemudian dapat dilihat koefisien korelasi antar variabel tersebut pada tabel 4.5.

Tabel 4.5. Koefisien korelasi dengan variabel dependen persentase FCaO

<i>Pearson Correlations</i>					
	Feed	Coal	Speed	Sekam	C3S
Feed	1	0,317	0,896	-0,069	-0,102
Coal	0,317	1	0,367	-0.161	-0.126
Speed	0,896	0,367	1	0,019	-0,045
Sekam	-0,069	-0.161	0,019	1	0.038
C3S	-0,102	-0.126	-0,045	0.038	1

Dapat dilihat pada tabel 4.5. bahwa nilai koefisien korelasi yang dihasilkan antara Feed dan Speed berada pada interval tingkat

korelasi yang sangat kuat. Sedangkan nilai koefisien korelasi yang dihasilkan antara Coal, Speed, Sekam, dan C3S masing-masing berada pada interval tingkat korelasi rendah dan sangat rendah. Sehingga, dapat disimpulkan variabel Feed, Coal, Speed dapat digunakan sebagai variabel masukan untuk melakukan prediksi persentase FCaO.

- Persentase C3S

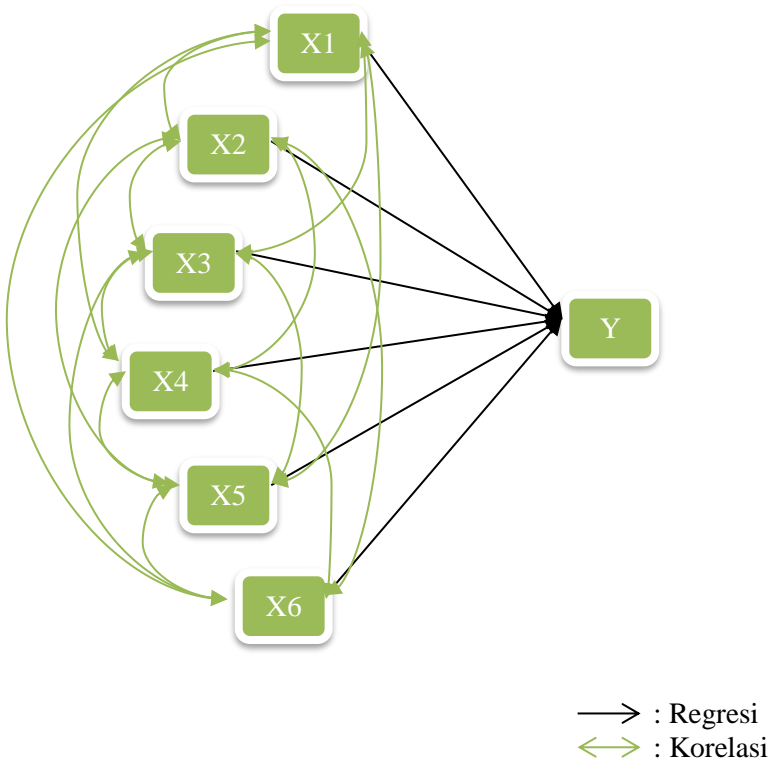
Terdapat 4 variabel yang memiliki pengaruh besar terhadap perubahan kecepatan angin dilihat dari hasil analisa regresi, yaitu Feed dengan koefisien beta -0,157, Coal dengan koefisien beta 0,176, Sekam dengan koefisien beta -0,135, FCaO dengan koefisien beta -0,338. Kemudian dapat dilihat koefisien korelasi antar variabel tersebut pada tabel 4.6.

Tabel 4.6. Koefisien korelasi dengan variabel dependen persentase C3S

<i>Pearson Correlations</i>				
	Feed	Coal	Sekam	FCaO
Feed	1	0,317	-0,069	0,068
Coal	0,317	1	-0,161	0,08
Sekam	-0,069	-0,161	1	-0,337
FCaO	0,068	0,08	-0,337	1

Dapat dilihat pada tabel 4.6. bahwa nilai koefisien korelasi yang dihasilkan antara Feed, Coal, Sekam, dan FCaO masing-masing berada pada interval tingkat korelasi rendah dan sangat rendah. Tetapi, dapat dilihat juga koefisien beta antara Feed dan Coal dengan persentase C3S bernilai cukup besar dibanding variabel lainnya. Sehingga, dapat disimpulkan variabel Feed dan Coal tetap dapat digunakan sebagai variabel masukan untuk melakukan prediksi persentase C3S.

4.1.3 Analisa Jalur (*Path Analysis*) Antar Variabel Cuaca



Gambar 4.1. Hipotesis awal diagram jalur antar variabel rotary kiln

Analisa jalur digunakan untuk untuk menunjukkan hubungan (korelasi) dan besar pengaruh (regresi) antar variabel yang berbentuk sebab akibat atau kausal yang dalam hal ini adalah variabel independen dan variabel dependen. Analisa jalur dilakukan dengan melihat hasil regresi dan korelasi yang telah dilakukan sebelumnya dan kemudian membuat hipotesis awal

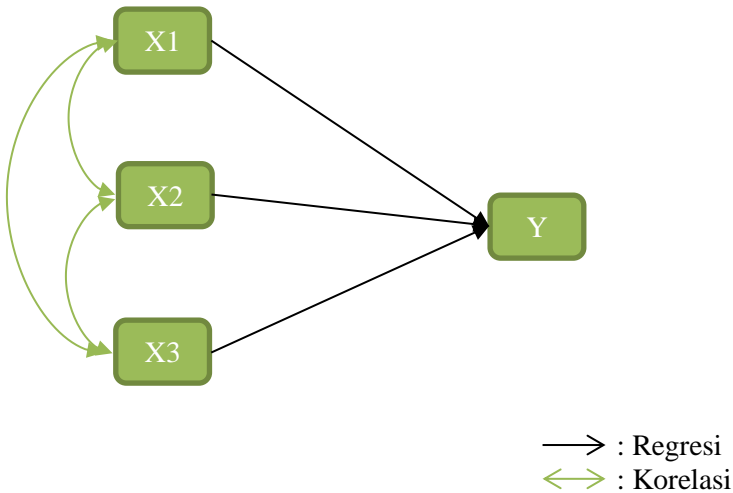
diagram jalur. Kemudian berdasarkan besar koefisien beta pada analisa regresi dan besar koefisien korelasi pada analisa korelasi, maka dapat dipilih variabel *rotary kiln* apa saja yang dapat digunakan sebagai masukan untuk melakukan persentase FCaO dan C3S.

- Persentase FCaO

Pada hipotesis awal diagram jalur untuk melakukan prediksi Persentase FCaO (Y), yang digunakan sebagai masukan atau dapat disebut juga variabel kausalnya adalah Feed (X1), Coal (X2), Speed (X3), Sekam (X4), Temp (X5), dan C3S (X6). Berikut ini merupakan kesimpulan dari hasil analisa jalur untuk variabel dependen persentase FCaO:

- a. Pengaruh variabel Feed terhadap FCaO adalah sebesar 0,382 atau 38,2%.
- b. Pengaruh variabel Coal terhadap FCaO adalah sebesar 0,107 atau 10,7%.
- c. Pengaruh variabel Speed terhadap FCaO adalah sebesar - 0,448 atau -44,8%.
- d. Pengaruh variabel independen secara gabungan terhadap variabel dependen FCaO adalah sebesar 0,249 atau 24,9%
- e. Pengaruh variabel lain di luar analisa jalur adalah sebesar 75,1%
- f. Korelasi antara variabel Feed dengan Coal adalah sebesar 0,317
- g. Korelasi antara variabel Feed dengan Speed adalah sebesar 0,896
- h. Korelasi antara variabel Coal dengan Speed adalah sebesar 0,367

Setelah dilakukan analisa jalur dengan menggunakan analisa regresi dan korelasi, maka diperoleh hasil diagram analisa jalur seperti pada gambar 4.2.



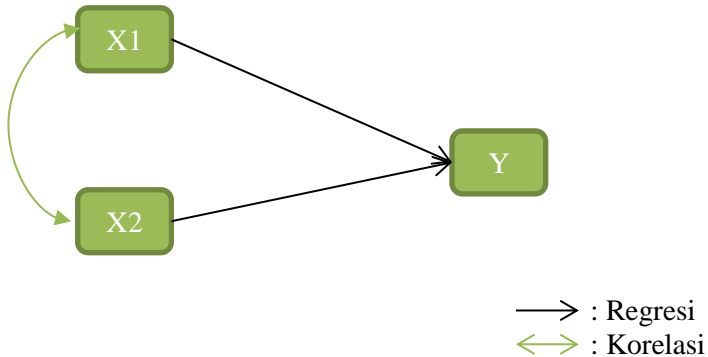
Gambar 4.2. Diagram jalur untuk prediksi persentase FCaO

- Persentase C3S

Pada hipotesis awal diagram jalur untuk melakukan prediksi Persentase C3S (Y), yang digunakan sebagai masukan atau dapat disebut juga variabel kausalnya adalah Feed (X1), Coal (X2), Speed (X3), Sekam (X4), Temp (X5), dan C3S (X6). Berikut ini merupakan kesimpulan dari hasil analisa jalur untuk variabel dependen kecepatan angin:

- a. Pengaruh variabel Feed terhadap C3S adalah sebesar - 0,157 atau 15,7%.
- b. Pengaruh variabel Coal terhadap C3S adalah sebesar 0,176 atau 17,6%.
- c. Pengaruh variabel independen secara gabungan terhadap variabel dependen C3S adalah sebesar 0,141 atau 14,1%.
- d. Pengaruh variabel lain di luar analisa jalur adalah sebesar 85,9%
- e. Korelasi antara variabel Feed dengan Coal adalah sebesar 0,317

Setelah dilakukan analisa jalur dengan menggunakan analisa regresi dan korelasi, maka diperoleh hasil diagram analisa jalur seperti pada gambar 4.3.

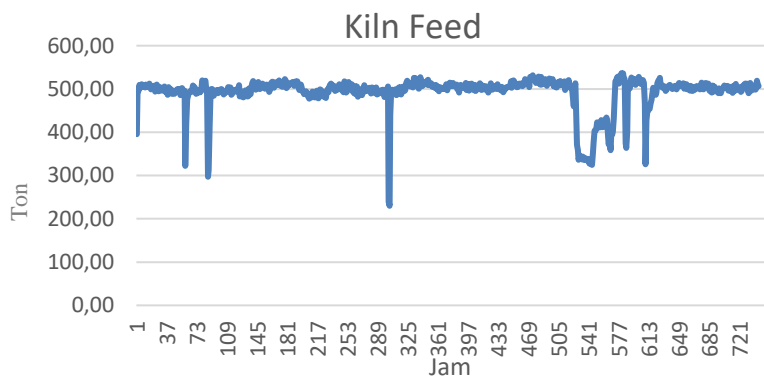


Gambar 4.3. Diagram jalur untuk prediksi persentase C3S

6.2. Prediksi Hasil Pembakaran Rotary Kiln dengan menggunakan Jaringan Syaraf Tiruan

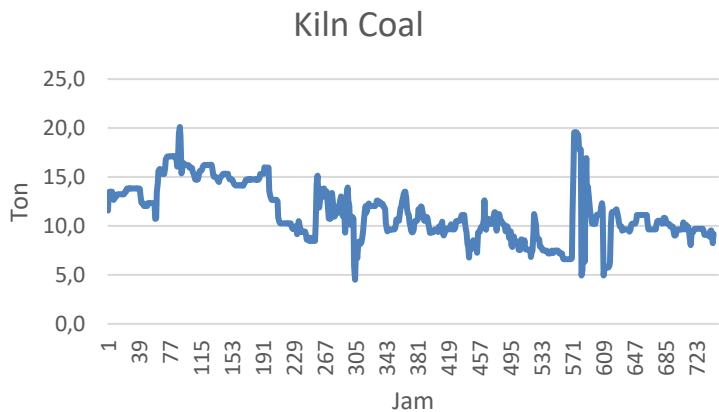
Jumlah *hidden node* pada arsitektur jaringan syaraf tiruan dapat mempengaruhi performa dan keakuratan dari hasil keluaran jaringan. Pada penelitian ini digunakan arsitektur jaringan syaraf tiruan dengan menggunakan satu *hidden layer*, satu *input layer*, dan satu *output layer*. Di dalam *hidden layer* terdapat beberapa *hidden node*, yang mana jumlah *hidden node* nantinya akan mempengaruhi hasil pelatihan dan validasi dari prediksi clinker yang akan dilakukan. Untuk mengetahui berapa jumlah *hidden node* yang akan menghasilkan nilai terbaik, dilakukan percobaan dengan menggunakan *hidden node* dengan jumlah 1 hingga 10 dengan menggunakan algoritma jaringan syaraf tiruan.

Adapun grafik data dari input dan output yaitu berupa *flow* dari *kiln feed*, *kiln coal*, sekam, *kiln speed*, FCaO dan C3S yang akan digunakan dapat dilihat pada gambar berikut:



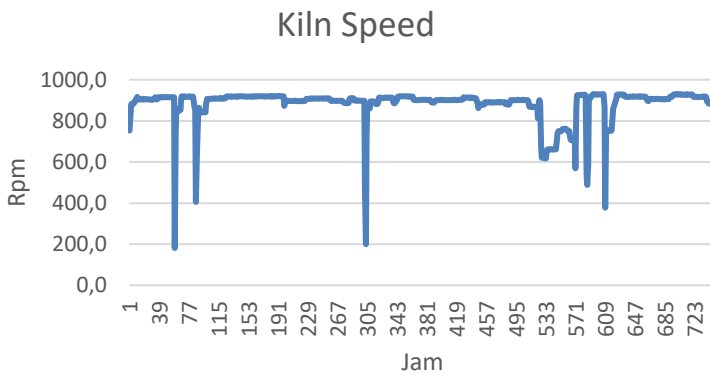
Gambar 4. 4 Grafik data *kiln feed*

Dari gambar 4.4 diatas menunjukkan *flow rate* kiln feed dalam ton selama 744 jam atau 1 bulan



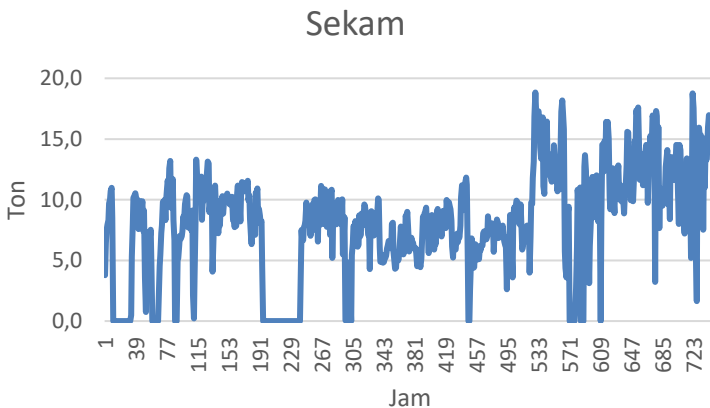
Gambar 4. 5 Grafik data *kiln coal*

Dari gambar 4.5 diatas menunjukkan *flow rate* kiln coal dalam ton selama 744 jam atau 1 bulan



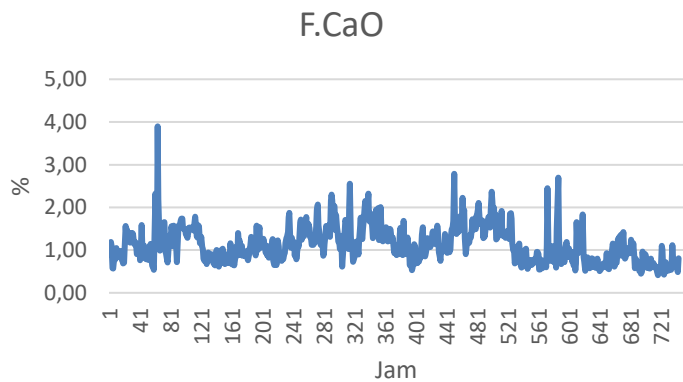
Gambar 4. 6 Grafik data *kiln speed*

Dari gambar 4.6 diatas menunjukkan kecepatan *kiln speed* dalam rpm selama 744 jam atau 1 bulan



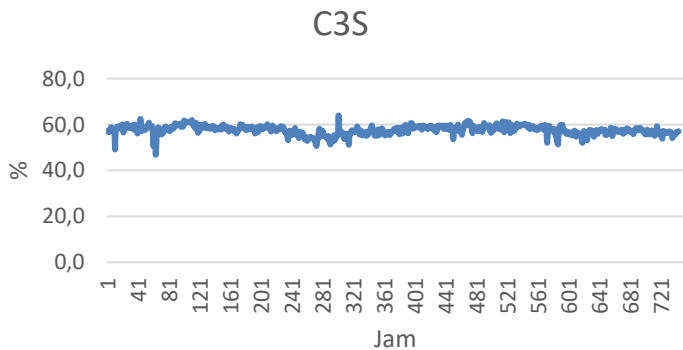
Gambar 4. 7 Grafik data sekam

Dari gambar 4.7 diatas menunjukkan *flow rate* sekam dalam ton selama 744 jam atau 1 bulan



Gambar 4. 8 Grafik data FCaO

Dari gambar 4.8 diatas menunjukkan *flow rate* FCaO dalam ton selama 744 jam atau 1 bulan



Gambar 4. 9 Grafik data C3S

Dari gambar 4.9 diatas menunjukkan *flow rate* C3S dalam ton selama 744 jam atau 1 bulan

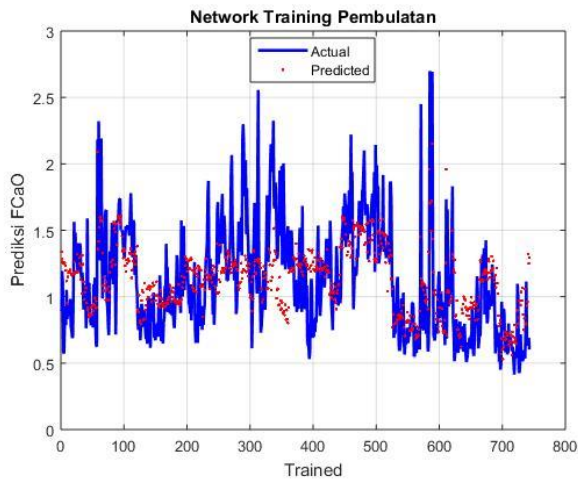
4.2.1 Jumlah *Hidden Node* untuk Memprediksi Persentase FCaO

Dari hasil pengujian data, yang dapat digunakan sebagai variabel masukan pada *input layer* untuk melakukan prediksi persentase FCaO adalah *kiln feed*, *kiln coal*, *kiln speed*, sekam dan, C3S. Hasil prediksi persentase FCaO menggunakan jaringan syaraf tiruan dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.7.

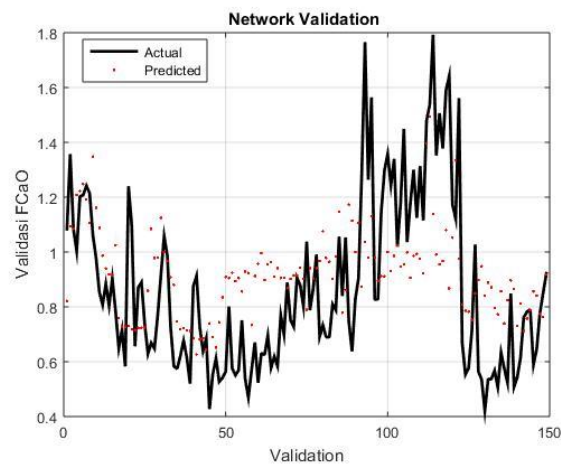
Tabel 4. 7 Nilai RMSE prediksi persentase FCaO dengan Mmenggunakan jaringan syaraf tiruan

Jumlah <i>Hidden Node</i>	Nilai RMSE
1	0,355727
2	0,312016
3	0,280895
4	0,272911
5	0,318696
6	0,269794
7	0,281506
8	0,319572
9	0,342689
10	0,356042

Dapat dilihat hasil prediksi berupa nilai RMSE, yang memiliki nilai terendah sebesar 0,269794 dengan jumlah *hidden node* yang digunakan sebanyak 6 *node*. Dari hasil prediksi tersebut kemudian diperoleh grafik hasil pelatihan data dan validasi data untuk prediksi jumlah clinker pada gambar 4.10 dan gambar 4.11.

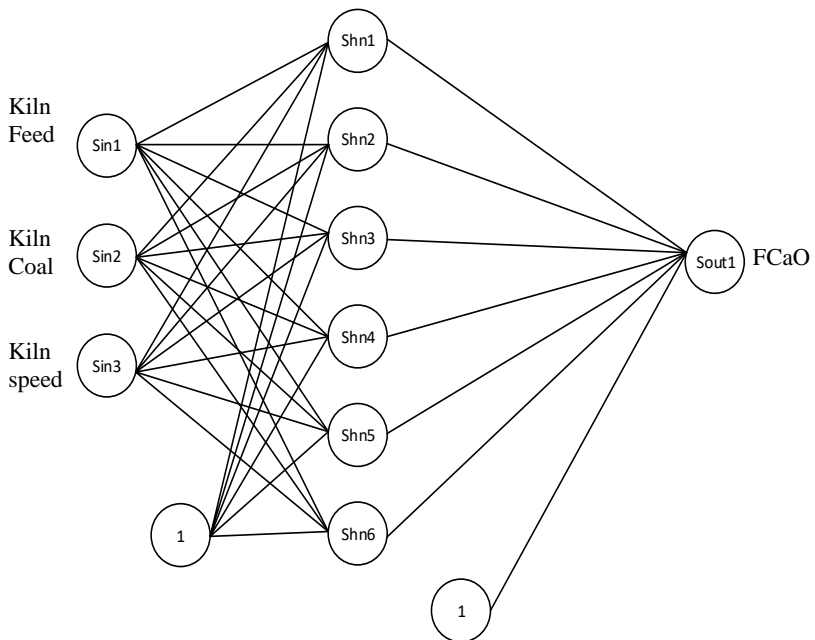


Gambar 4. 10 Hasil pelatihan data persentase FCaO



Gambar 4. 11 Hasil validasi data persentase FCaO

Dari hasil pengujian data telah diperoleh variabel masukan yang tepat untuk melakukan prediksi FCaO. Sehingga diperoleh masukan jaringan syaraf tiruan untuk prediksi FCaO berjumlah 3 variabel yaitu Kiln Feed, Kiln Coal, dan Kiln Speed. Selain itu, untuk jumlah variabel diambil dari hasil pengujian jumlah *hidden node* yang memiliki nilai RMSE terkecil yaitu pada *hidden node* yang berjumlah 6 *node* pada FCaO, Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah. seperti pada gambar 4.12



Gambar 4. 12 Arsitektur pada metode JST prediksi FCaO

4.2.2 Jumlah *Hidden Node* untuk Memprediksi Persentase C3S

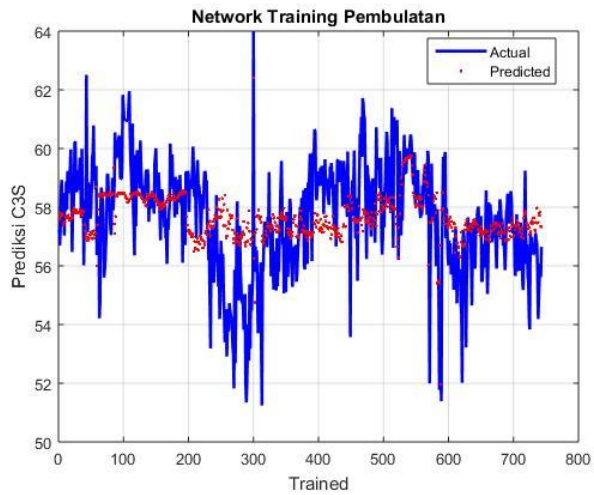
Dari hasil pengujian data, yang dapat digunakan sebagai variabel masukan pada *input layer* untuk melakukan prediksi

persentase C3S adalah kiln feed, kiln coal, sekam, dan FCaO. Hasil prediksi persentase C3S menggunakan jaringan syaraf tiruan dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.8.

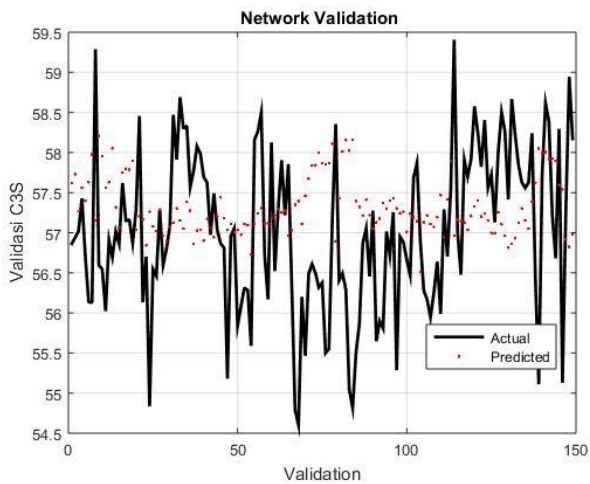
Tabel 4.8 Nilai RMSE prediksi persentase C3S dengan menggunakan jaringan syaraf tiruan

Jumlah <i>Hidden Node</i>	Nilai RMSE
1	1,778374
2	1,2073
3	1,176368
4	1,174429
5	1,224774
6	1,209307
7	1,19665
8	1,273225
9	1,312022
10	1,275266

Dapat dilihat hasil prediksi berupa nilai RMSE, yang memiliki nilai terendah sebesar 1,174429 dengan jumlah *hidden node* yang digunakan sebanyak 4 *node*. Dari hasil prediksi tersebut kemudian diperoleh grafik hasil pelatihan data dan validasi data untuk prediksi jumlah clinker pada gambar 4.13 dan gambar 4.14.

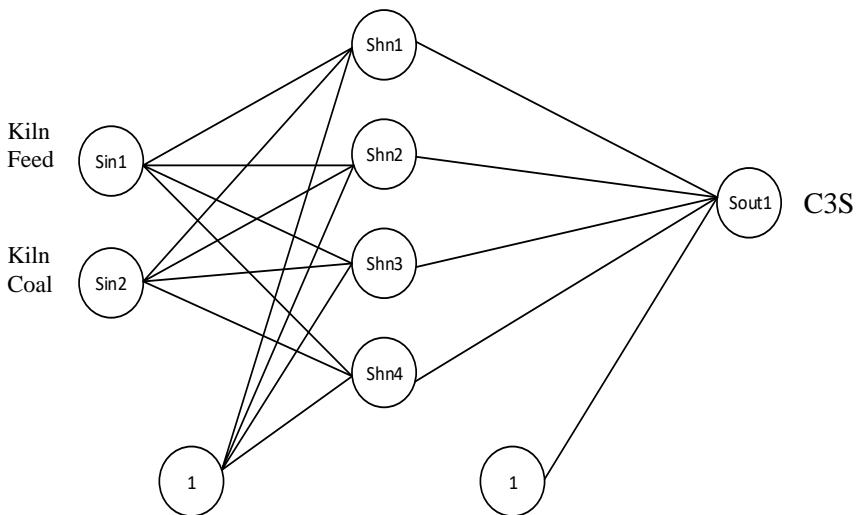


Gambar 4. 13 Hasil pelatihan data persentase C3S



Gambar 4. 14 Hasil validasi data persentase C3S

Dari hasil pengujian data telah diperoleh variabel masukan yang tepat untuk melakukan prediksi C3S. Sehingga diperoleh masukan jaringan syaraf tiruan untuk prediksi C3S berjumlah 2 variabel yaitu Kiln Feed dan Kiln Coal. Selain itu, untuk jumlah variabel diambil dari hasil pengujian jumlah *hidden node* yang memiliki nilai RMSE terkecil yaitu pada *hidden node* yang *hidden node* 4 pada C3S, Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah. seperti pada gambar 4.15



Gambar 4. 15 Arsitektur pada metode JST prediksi C3S

6.3. Prediksi Hasil Pembakaran Rotary Kiln dengan menggunakan Jaringan Syaraf Tiruan dan Optimisasi Genetic Algorithm

Setelah melakukan prediksi hasil pembakaran *rotary kiln* menggunakan metode jaringan syaraf tiruan dengan hasil yang dapat dilihat pada sub bab sebelumnya, berikutnya dilakukan percobaan prediksi cuaca dengan menggunakan metode gabungan jaringan syaraf tiruan dan optimisasi *genetic algorithm*. Hal ini

dilakukan untuk mengetahui apakah metode gabungan jaringan syaraf tiruan dan optimisasi *genetic algorithm* dapat meningkatkan akurasi dari prediksi hasil pembakaran *rotary kiln* dengan melihat nilai RMSE yang dihasilkan. Percobaan dilakukan dengan menggunakan variasi jumlah *hidden node* yang sama dengan metode jaringan syaraf tiruan yaitu berjumlah 1 hingga 10 *node*.

4.3.1 Persentase FCaO

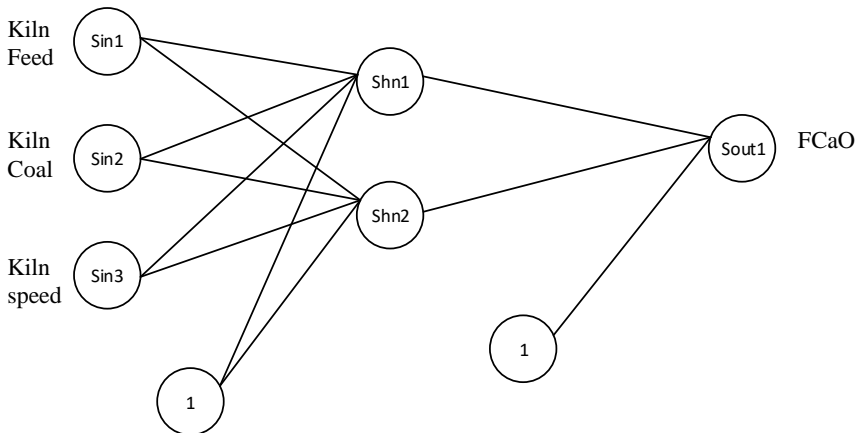
Variabel masukan yang digunakan pada percobaan ini tetap menggunakan hasil dari pengujian data, yaitu menggunakan variabel *kiln feed*, *kiln coal*, dan *kiln speed*, untuk melakukan prediksi persentase FCaO. Hasil prediksi persentase FCaO dengan menggunakan metode gabungan jaringan syaraf tiruan dan optimisasi *genetic algorithm* dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.9

Tabel 4.9 Nilai RMSE prediksi persentase FCaO dengan menggunakan jaringan syaraf tiruan dan optimisasi *genetic algorithm*

Jumlah <i>Hidden Node</i>	Nilai RMSE
1	0.0956
2	0.0948
3	0.0976
4	0.1019
5	0,0954
6	0,1020
7	0,0974
8	0,1061
9	0,1034
10	0,1071

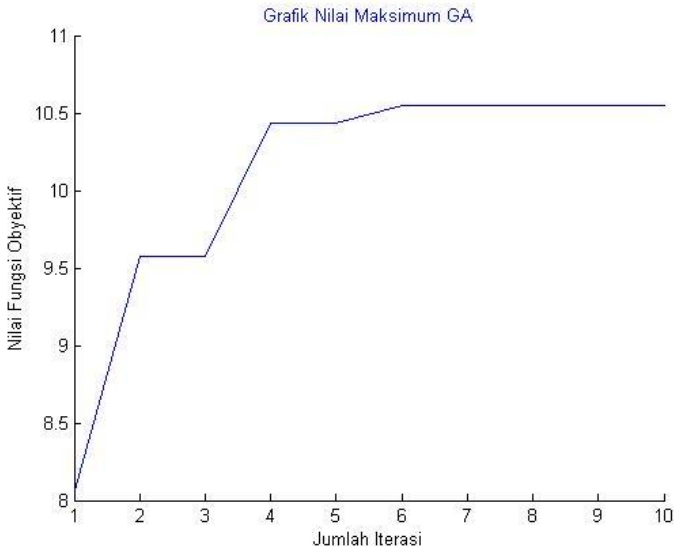
Dapat dilihat pada tabel 4.9, hasil prediksi yang berupa nilai RMSE. Nilai RMSE terendah yaitu sebesar 0.0948 terdapat pada arsitektur jaringan syaraf tiruan dengan jumlah *hidden node* sebanyak 2 *node*. Jika dibandingkan dengan metode tanpa optimisasi, nilai RMSE yang dihasilkan oleh metode gabungan jauh lebih kecil. Dari hal ini dapat disimpulkan bahwa metode gabungan jaringan syaraf tiruan dan optimisasi *genetic algorithm* dapat digunakan untuk melakukan prediksi hasil pembakaran rotary kiln dengan hasil yang lebih akurat dibandingkan dengan metode tanpa optimisasi.

Berikut ini adalah Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah. seperti pada gambar 4.16



Gambar 4. 16 Arsitektur pada metode JST-GA prediksi FCaO

Grafik hasil optimisasi menggunakan Optimisasi *Genetic Algorithm* dapat dilihat pada gambar 4.17.



Gambar 4. 17 Grafik optimisasi pada prediksi persentase FCaO

Grafik pada gambar 4.17 menunjukkan bahwa nilai bobot yang terdapat pada jaringan syaraf tiruan dapat di perbaharui menjadi lebih optimum dan akan menghasilkan nilai fungsi obyektif yang semakin besar seiring dengan bertambahnya iterasi.

4.3.2 Persentase C3S

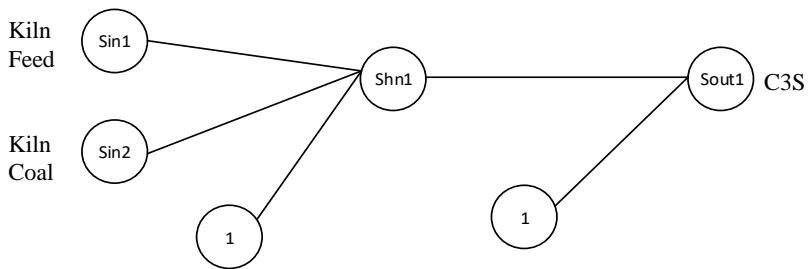
Variabel masukan yang digunakan pada percobaan ini tetap menggunakan hasil dari pengujian data, yaitu menggunakan variabel *kiln feed* dan *kiln coal* untuk melakukan prediksi presentase C3S. Hasil prediksi persentase C3S dengan menggunakan metode gabungan jaringan syaraf tiruan dan optimisasi *genetic algorithm* dengan variasi jumlah *hidden node* dapat dilihat pada tabel 4.10.

Tabel 4.10 Nilai RMSE prediksi persentase C3S dengan menggunakan jaringan syaraf tiruan dan optimisasi *genetic algorithm*

Jumlah <i>Hidden</i> <i>Node</i>	Nilai RMSE
1	0.0878
2	0,0887
3	0,0923
4	0,0935
5	0,0918
6	0.0879
7	0.0878
8	0.0916
9	0.0888
10	0.0887

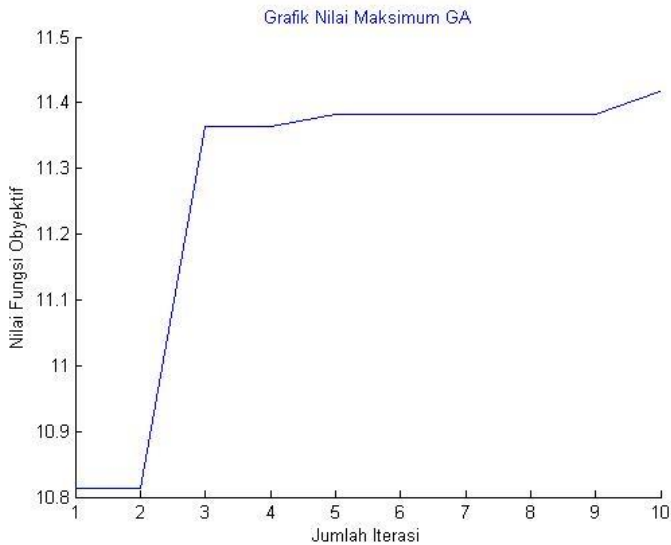
Dapat dilihat pada tabel 4.10, hasil prediksi yang berupa nilai RMSE. Nilai RMSE terendah yaitu sebesar 0.0878 terdapat pada arsitektur jaringan syaraf tiruan dengan jumlah *hidden node* sebanyak 1 *node*. Jika dibandingkan dengan metode tanpa optimisasi, nilai RMSE yang dihasilkan oleh metode gabungan jauh lebih kecil. Dari hal ini dapat disimpulkan bahwa metode gabungan jaringan syaraf tiruan dan optimisasi *genetic algorithm* dapat digunakan untuk melakukan prediksi hasil pembakaran rotary kiln dengan hasil yang lebih akurat dibandingkan dengan metode tanpa optimisasi.

Berikut ini adalah Arsitektur yang dipilih adalah jumlah *hidden node* yang menghasilkan nilai RMSE terendah. seperti pada gambar 4.18



Gambar 4. 18 Arsitektur pada metode JST-GA prediksi FCaO

Grafik hasil optimisasi menggunakan Optimisasi *Genetic Algorithm* dapat dilihat pada gambar 4.19.



Gambar 4. 19 Grafik optimisasi pada prediksi persentase C3S

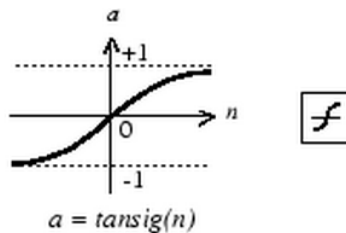
Grafik pada gambar 4.19 menunjukkan bahwa nilai bobot yang terdapat pada jaringan syaraf tiruan dapat di perbaharui menjadi lebih optimum dan akan menghasilkan nilai fungsi obyektif yang semakin besar seiring dengan bertambahnya iterasi.

4.4 Parameter Arsitektur Jaringan Syaraf Tiruan dari Hasil Perancangan

4.4.1 Fungsi Aktivasi

Fungsi aktivasi digunakan untuk menghasilkan nilai keluaran dari setiap node yang dilewati. Terdapat beberapa jenis fungsi aktivasi yang paling sering digunakan pada jaringan syaraf tiruan yaitu sigmoid logaritmik (*logsig*), sigmoid tangen (*tansig*), dan linear (*purelin*). Fungsi aktivasi sigmoid logaritmik dan sigmoid tangen dapat meminimalisir kapasitas komputasi untuk pelatihan, sedangkan fungsi aktivasi linear lebih banyak digunakan pada *output node* (Montesdeoca-Contreras, Zambrano-Abad, Morales-Garcia, & Ávila-Campoverde, 2014).

Pada penelitian ini, fungsi aktivasi diaplikasikan pada setiap *hidden node* *ouput node* Fungsi aktivasi yang digunakan adalah sigmoid tangen seperti pada gambar dibawah ini.



Gambar 4. 20.Fungsi aktivasi sigmoid tangen pada JST

Sigmoid tangen digunakan pada penelitian ini, karena pada penelitian yang sudah ada telah dilakukan perbandingan antara kombinasi fungsi aktivasi antara sigmoid logaritmik (*logsig*), sigmoid tangen (*tansig*), dan linear (*purelin*) dengan hasil kombinasi sigmoid tangen pada *hidden node* dan *output node* memiliki hasil yang terbaik (Choudhary, Rishi, Ahlawat, & Dhaka,

2010). Hal tersebut juga telah dibuktikan dengan hasil pada penelitian tugas akhir ini yang mana semua nilai RMSE yang dihasilkan pada prediksi presentase FCaO dan C3S dengan dua metode yang berbeda menghasilkan nilai yang mendekati nilai nol yang menunjukkan akurasi yang baik.

4.4.2 Faktor Bobot Optimum

Algoritma pelatihan yang digunakan pada prediksi FCaO dan C3S dengan metode jaringan syaraf tiruan adalah algoritma Levenberg-Marquardt, sedangkan pada metode gabungan menggunakan algoritma *genetic algorithm*. Algoritma pelatihan yang digunakan pada metode gabungan dapat menghasilkan bobot yang optimum dengan nilai RMSE yang tidak jauh berbeda dengan metode jaringan syaraf tiruan.

Faktor bobot dapat disimbolkan dengan huruf v_{ij} untuk faktor bobot yang masuk ke *hidden node* dan w_j untuk faktor bobot yang masuk ke *output node*. Huruf i dan j secara berurutan menunjukkan penamaan untuk *input node* dan *hidden node*. Sehingga, v_{ij} menunjukkan faktor bobot yang berasal dari *input node* ke- i menuju *hidden node* ke- j , sedangkan w_j menunjukkan faktor bobot yang berasal dari *hidden node* ke- j .

Nilai faktor bobot yang dihasilkan dari prediksi Persentase FCaO dengan metode jaringan syaraf tiruan dengan algoritma pelatihan Levenberg-Marquardt dapat dilihat pada tabel 4.11.

Tabel 4. 11 Faktor bobot optimum dari metode JST untuk prediksi persentase FCaO

Posisi Bobot	Nilai Bobot	Posisi Bobot	Nilai Bobot
v_{11}	4,72885	v_{15}	0,66294
v_{21}	0,39452	v_{25}	-1,95589
v_{31}	-6,14056	v_{35}	-3,57784
v_{01}	1,91023	v_{05}	3,68827
v_{12}	8,85622	v_{16}	37,8145
v_{22}	14,4866	v_{26}	-15,9876
v_{32}	6,99378	v_{36}	-15,0437

Posisi Bobot	Nilai Bobot	Posisi Bobot	Nilai Bobot
v_{02}	-6,95350	v_{06}	-8,14959
v_{13}	-0,19269	w_1	0,37144
v_{23}	-0,26321	w_2	-0,35862
v_{33}	26,1641	w_3	-24,3762
v_{03}	-24,9631	w_4	-0,49506
v_{14}	-1,55647	w_5	0,46484
v_{24}	0,94851	w_6	-1,32855
v_{34}	-4,85233	w_0	-1,28209
v_{04}	5,47465		

Bobot yang dihasilkan pada prediksi persentase FCaO menggunakan metode jaringan syaraf tiruan berjumlah 31 disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 6 *node*.

Nilai faktor bobot yang dihasilkan dari prediksi persentase C3S dengan metode jaringan syaraf tiruan dengan algoritma pelatihan *Levenberg-Marquardt* dapat dilihat pada tabel 4.12.

Tabel 4.12. Faktor bobot optimum dari metode JST untuk prediksi persentase C3S

Posisi Bobot	Nilai Bobot	Posisi Bobot	Nilai Bobot
v_{11}	-10,6654	v_{14}	-0,11272
v_{21}	8,78422	v_{24}	2,06323
v_{01}	4,67777	v_{04}	0,57525
v_{12}	3,66595	w_1	0,09926
v_{22}	-2,84903	w_2	0,05202
v_{02}	-3,68084	w_3	-1,19538
v_{13}	-0,86864	w_4	-174,869
v_{23}	0,01254	w_0	14,7536
v_{03}	1,98832		

Bobot yang dihasilkan pada prediksi persentase C3S menggunakan metode jaringan syaraf tiruan berjumlah 17

disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 4 *node*.

Nilai faktor bobot yang dihasilkan dari prediksi persentase FCaO angin dengan metode gabungan JST-GA dengan algoritma pelatihan yang digunakan adalah *genetic algorithm* dapat dilihat pada tabel 4.13.

Tabel 4.13 Faktor bobot optimum dari metode JST-GA untuk prediksi persentase FCaO

Posisi Bobot	Nilai Bobot	Posisi Bobot	Nilai Bobot
v_{11}	0,686788	v_{32}	0,067318
v_{21}	0,533320	v_{02}	-0,176642
v_{31}	-0,74024	w_1	0,336598
v_{01}	-0,046374	w_2	0,0440245
v_{12}	0,3512395	w_0	0,1788150
v_{22}	0,7550065		

Bobot yang dihasilkan pada prediksi persentase FCaO menggunakan metode jaringan syaraf tiruan berjumlah 11 disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 2 *node*.

Nilai faktor bobot yang dihasilkan dari prediksi persentase C3S dengan metode gabungan JST-GA dengan algoritma pelatihan yang digunakan adalah *genetic algorithm* dapat dilihat pada tabel 4.14.

Tabel 4.14 Faktor bobot optimum dari metode JST-GA untuk prediksi persentase C3S

Posisi Bobot	Nilai Bobot	Posisi Bobot	Nilai Bobot
v_{11}	-0,93292	w_1	0,126336
v_{21}	-0,317351	w_0	0,737819
v_{01}	0,525214		

Bobot yang dihasilkan pada prediksi persentase C3S menggunakan metode jaringan syaraf tiruan berjumlah 5 disesuaikan dengan banyaknya *hidden node* yang digunakan yaitu 1 *node*.

4.5 Perbandingan Hasil Antara Kedua Metode

Dari percobaan yang telah dilakukan dengan menggunakan metode jaringan syaraf tiruan dan metode gabungan jaringan syaraf tiruan dan *genetic algorithm* dengan menggunakan variasi *hidden node*, diperoleh hasil terbaik yang dapat dibandingkan yaitu nilai RMSE prediksi terkecil, jumlah *hidden node* terbaik, dan iterasi pada saat diperolehnya nilai RMSE tersebut.

Tabel 4.15 Perbandingan hasil terbaik antara metode JST dan metode gabungan JST-GA

Variabel		JST	JST-GA
FCaO	Nilai RMSE	0,269794	0.0948
	Jumlah <i>Hidden Node</i>	6	2
	Epoch	97	6
C3S	Nilai RMSE	1,174429	0.0878
	Jumlah <i>Hidden Node</i>	4	1
	Epoch	142	10

Dari tabel 4.5. dapat dilihat perbandingan performa antara kedua metode tersebut yang dibagi menjadi dua prediksi yaitu prediksi persentase FCaO dan C3S, dan terdapat dua metode yaitu metode jaringan syaraf tiruan (JST) dan metode gabungan jaringan syaraf tiruan dan *genetic algorithm* (JST-GA). Parameter yang dapat dibandingkan antara kedua metode prediksi tersebut yaitu jumlah iterasi untuk memperoleh nilai RMSE yang sama dengan menggunakan jumlah *hidden node* yang sama.

Pada prediksi FCaO, Metode JST yang digunakan untuk melakukan prediksi FCaO dapat menghasilkan nilai RMSE

0,269794 pada epoch ke -97 dan hidden node 6, sedangkan metode JST-GA menghasilkan 0.0948 pada epoch ke-6 dan hidden node 2. Dari hal ini dapat disimpulkan bahwa metode JST-GA memiliki performa yang baik dalam melakukan prediksi FCaO, dan dapat menghasilkan bobot yang optimum karena nilai RMSE yang dihasilkan mendekati nol.

Pada prediksi C3S, metode JST dapat menghasilkan nilai RMSE sebesar 1,174429 pada epoch ke-142 dan hidden node 4, sedangkan metode JST-GA menghasilkan nilai RMSE sebesar 0.0878 pada epoch ke-10 dan hidden node 1. Dari hal ini dapat disimpulkan bahwa metode JST-GA memiliki performa yang baik dalam melakukan prediksi FCaO, dan dapat menghasilkan bobot yang optimum karena nilai RMSE yang dihasilkan mendekati nol.

“Halaman ini sengaja dikosongkan”

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Terdapat beberapa kesimpulan yang diperoleh dari penelitian tugas akhir ini, yaitu sebagai berikut:

- Pada model prediktor dengan menggunakan jaringan syaraf tiruan berhasil mendapatkan Nilai RMSE terbaik yaitu terdapat pada *hidden node* berjumlah 6 *node* pada prediksi persentase FCaO yaitu sebesar 0,269794 dan pada epoch ke-97, dan 4 *node* pada prediksi persentase C3S yaitu sebesar 1,174429 pada epoch ke-142.
- Pada model prediktor dengan menggunakan jaringan syaraf tiruan dan optimisasi *genetic algorithm* berhasil mendapatkan nilai RMSE terbaik yaitu terdapat pada *hidden node* berjumlah 2 *node* pada prediksi persentase FCaO yaitu sebesar 0.0948 pada epoch ke-6, dan 1 *node* pada prediksi Persentase C3S yaitu sebesar 0.0878 pada epoch ke-10.

5.2. Saran

Saran yang dapat diajukan terkait dengan penelitian tugas akhir ini, antara lain:

- a. Pada penelitian selanjutnya dapat dilakukan percobaan dengan menggunakan metode optimisasi selain Optimisasi *Genetic Algorithm*. untuk melihat apakah semua metode optimisasi dapat memberikan hasil yang lebih baik dibandingkan tanpa menggunakan metode optimisasi.
- b. Sebaiknya dilakukan juga penelitian dengan memperhatikan parameter lain yang berpengaruh terhadap hasil pembakaran di *rotary kiln* sehingga didapatkan hasil prediksi yang lebih akurat

“Halaman ini sengaja dikosongkan”

DAFTAR PUSTAKA

- Athul, M., & S, A. (2016). Artificial Neural Network Model For Heat Transfer Process In a Cement Kiln . *TJPRC: International Journal of Power Systems & Microelectronics (TJPRC:IJPSM)*, II(2), 1-10.
- Atiliani, A. (2013). *Pelatihan Jaringan Syaraf Tiruan Multilayer Perceptron Menggunakan Genetic Algorithm Levenberg Marquardt*. Surakarta: Perpustakaan Universitas Sebelas Maret.
- Ayu, R., Hadi, H., & Sarwono. (2016). Audit Energi Pada Sistem Rotary Kiln System di Pabrik Semen.
- Boateng, A. (2008). *Rotary Kilns Transport Phenomena and Transport Processes*. Oxford: Elsevier Inc.
- BPPT. (2013). Pengembangan Energi dalam Mendukung Sektor Transportasi dan Industri Pengolahan Mineral. In *Outlook Energi Indonesia 2013* (pp. 1-116). Jakarta: Pusat Teknologi Pengembangan Sumberdaya Energi (PTPSE).
- Choudhary, A., Rishi, R., Ahlawat, S., & Dhaka, V. S. (2010). Performance Analysis of Feed Forward MLP with various Activation Functions for Handwritten Numerals Recognition. *The 2nd International Conference on Computer and Automation Engineering (ICCAE)*. 5, pp. 852-856. Singapore: IEEE.
- Demuth, H., Beale, M., & Hagan, M. (2004). *Neural Network Toolbox™ 6 User's Guide*. Natick, Massachusetts: The MathWorks, Inc.
- ESDM. (n.d.). *Energi Baru Terbarukan*. Retrieved Maret 28, 2016, from www.esdm.go.id
- Fausett, L. (n.d.). *Fundamentals Of Neural Network "Architectures, Algorithms and Applications"*.
- Gill, E. J., Singh, E. B., & Singh, E. S. (2010). Training Back Propagation Neural Networks with Genetic Algorithm for Weather Forecasting. *IEEE 8th International Symposium on Intelligent Systems and Informatics*, (pp. 465-469). Serbia.

- Institut Teknologi Sepuluh Nopember (ITS). (n.d.). *Jaringan Syaraf Tiruan*. Retrieved January 31, 2017, from <http://share.its.ac.id>
- Kementerian Energi dan Sumber Daya Mineral. (n.d.). Retrieved January 15, 2017, from <http://www.esdm.go.id/>
- Litta A. J, S. M. (2012). Artificial Neural Network Model for the Prediction of Thunderstorms over Kolkata. *International Journal of Computer Applications*, 50, 50-55.
- Madlool, N., & al, e. (2011). *A Critical Review on Energy Use and Savings in the Cements Industries*.
- Mokosuli, L., Weku, W., & Latumakulita, L. (n.d.). Prediksi Tingkat Kriminalitas Menggunakan Jaringan Syaraf Tiruan Backpropagation : Algoritma Lavenberg Marquardt di Kota Manado Berbasis Sistem Informasi Geografi.
- Montesdeoca-Contreras, J. C., Zambrano-Abad, J. C., Morales-Garcia, J. A., & Ávila-Campoverde, R. S. (2014). Virtual Speed Sensor for DC Motor using Back-Propagation Artificial Neural Networks. *IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*. Ixtapa, Mexico: IEEE.
- Nørskov, L. K., Dam-Johansen, K., Glarborg, P., Jensen, P. A., & Larsen, M. B. (2012). Combustion of Solid Alternative Fuels in The Cement Kiln Burner.
- Pazand, K., Panahi, M. S., & Pourabdoli, M. (2009). Simulating the Mechanical of a Rotary Cement Kiln Using Artificial Neural Network. *Material and Design* (30), 3468-3473.
- Peray, K. (1979). *Cement Manufacturer's Handbook*. New York, NY: Chemical Publishing Co.Inc.
- Politeknik Elektronika Negeri Surabaya (PENS). (n.d.). *Ulasan Algoritma Genetika* . Retrieved December 22, 2016, from <http://entin.lecturer.pens.ac.id>
- Priyatno, D. (2009). *5 Jam Belajar Olah Data dengan SPSS 17*. (J. Widiyatmoko, Ed.) Yogyakarta, Indonesia: C.V ANDI OFFSET.

- Puslitbang Teknologi Mineral dan Batubara. (2011). Jakarta, Indonesia.
- Puspitorini, R. W., Damayanti, I. S., Nurtono, T., & Winardi, S. (2013). Studi Pemanfaatan Batubara Kualitas Rendah Sebagai Bahan Bakar Rotary Cement Kiln Berbasis CFD. *Jurnal Teknik POMITS, II*.
- Rao, S. S. (2009). Engineering Optimization: Theory and Practice . John Wiley & Sons.
- Santoso, S. (2010). *Statistik Multivariat Konsep dan Aplikasi dengan SPSS*. Kompas Gramedia.
- Sugiyono. (2017). *Statistika Untuk Penelitian*. Bandung, Jawa Barat, Indonesia: Alfabeta.
- Teja, R., P, S., & M, a. G. (2016). Control and Optimization of a Triple String Rotary Cement Kiln using Model Predictive Control. *IFAC-PaperOnLine 49-1*, 748-753.
- Wahyu, D., & Sumiati, R. (n.d.). Analisis Energi pada Sistem Rotary Kiln Unit Indarung IV, PT. Semen Padang.
- Walter, H. D. (1985). Cement Data Book. London: Macdonald & Even.
- Widhiyasa, A. (n.d.). Kajian Genetic Alghorithm Dalam Penyelesaian TSP.
- Yin, Q., Du, W.-J., Ji, X.-L., & Cheng, a. L. (2016). Design Requirements and Performance Optimization of Waste Heat Recovery Systems for Rotary Kiln. *International Journal of Heat and Mass Transfer 94*(93), 1-8.
- Yin, Q., Du, W.-J., Ji, X.-L., & Cheng, a. L. (2016). Optimization Design and Economic Analyses of Heat Recovery Exchangers on Rotary Kiln. *Applied Energy*(180), 743-756.

“Halaman ini sengaja dikosongkan”

LAMPIRAN

Lampiran 1. Koding Jaringan Syaraf Tiruan

```
clc
close all;
clear all;
clc
disp('-----')
disp('  TRAINING IN PROGRESS  ')
disp('-----')
A = xlsread('E:\Kuliah\Road to S.T\Semester
8\TA\JST\JSTdanGA.xlsx','Kiln3');
% Panggil data
% [rowTr,colTr] = size(A);

ut = A(1:744,1:4)'; %input di transpose
yt = A(1:744,6)'; % output di transpose

uv = A(745:893,1:4)'; %input di transpose
yv = A(745:893,6)'; %output di transpose

us = A(1:893,1:4)'; %input di transpose
ys = A(1:893,6)'; %output ditranspose

[rowv,colv] = size(uv);
[rowu,colu] = size(ut);
[rowy,coly] = size(yt);
Min = -ones(rowu,1);
Max = ones(rowu,1);
MM = [Min Max];

for i=1:rowu
    maxusa(i)=max(us(i,:)); %change the range
    minusa(i)=min(us(i,:));
end
```

```

for i = 1:rowy
    maxys(i)=max(ys(i,:));
    minys(i)=min(ys(i,:));
end

```

```

minmaxus = [maxusa;minusa];
minmaxys = [maxys;minys];

```

```

for i = 1:rowy
    yt(i,:)=((2/(max(ys(i,:))-
min(ys(i,:))))*(yt(i,:)-min(ys(i,:)))-1;
    yv(i,:)=((2/(max(ys(i,:))-
min(ys(i,:))))*(yv(i,:)-min(ys(i,:)))-1;
end

```

```

for j = 1:colu
    for i=1:rowu
        ut(i,j)=((2/(maxusa(i)-
minusa(i))))*(ut(i,j)-minusa(i)))-1;
    end
end

```

```

for j = 1:colv
    for i=1:rowv
        uv(i,j)=((2/(maxusa(i)-
minusa(i))))*(uv(i,j)-minusa(i)))-1;
    end
end

```

```

ut1=ut(1,:)';
ut2=ut(2,:)';
ut3=ut(3,:)';
ut4=ut(4,:)';

```

```

yt1 = yt(1,:)';

```

```

uv1=uv(1,:)';
uv2=uv(2,:)';

```



```

uv3=uv(3,:)' ;
uv4=uv(4,:)' ;

yv1 = yv(1,:)' ;

% history length for MIMO identification
hist = ones(1,4);

%hist = [1 1 1 1 2 2 2 2 3 3 3 3 1 1];

[n_rows,n_col] = size(ut1);

% setting training data matrix
data_latih = zeros(n_rows-1,sum(hist));

for i = 1:hist(1),
    data_latih(:,i) = [zeros(hist(1)-
i,1);ut1(2:n_rows-hist(1)+i)];
end

for j = 1:hist(2),
    data_latih(:,sum(hist(1))+j) =
[zeros(hist(2)-j,1);ut2(2:n_rows-hist(2)+j)];
end

for k = 1:hist(3),
    data_latih(:,sum(hist(1:2))+k) =
[zeros(hist(3)-k,1);ut3(2:n_rows-hist(3)+k)];
end

for l = 1:hist(4),
    data_latih(:,sum(hist(1:3))+l) =
[zeros(hist(4)-l,1);ut4(2:n_rows-hist(4)+l)];
end

PHI = data_latih';

```

```

% Construction of output matrix
Y = zeros(n_rows-1,1);
Y(:,1) = yt1(2:end);

Ys = Y';

%coba
% Construction of networks structure
NetDef = [];
netdef1 = 'H';
netdef2 = 'L';
L = [netdef1;netdef2];
Data_RMSE =[];
trparms = settrain;

for x=1:10;
    hn = x;
    close all;
    Ys = Y';
    NetDef = [NetDef L];
    netdef1 = 'H';
    netdef2 = '-';
    L = [netdef1;netdef2];

    % Construction of networks structure

    trparms = settrain;

    [W1,W2,PI_vec,yhat] =
marq_rev(NetDef, [], [], PHI, Ys, trparms);

    % RMSE calculation

    for i = 1:1

RMSE_train(i)=r_m_s_e(yhat(i,:),Ys(i,:));

```

```

end

for i = 1:1
    Ys(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(Ys(i, :)+1)/2)+min(ys(i, :));
%Descaling
    Yhat(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(yhat(i, :)+1)/2)+min(ys(i, :));
%Descaling
    RMSE_train_f(i) =
r_m_s_e(Ys(i,:),Yhat(i,:));

end

%Drawing
for i = 1
    figure(i)
    plot(Ys(i,:), 'b', 'LineWidth', 2);
    hold on
    plot(Yhat(i,:), 'r.', 'LineWidth', 1);
    grid
    title('Network Training Pembulatan');
    legend('Solid : Actual', 'Dot : Predicted
', 'Location', 'Best');
    ylabel('Prediksi Clinker');
    xlabel('Trained');
end

save WT_Cat NetDef W1 W2 maxys minys maxusa
minusa
xlswrite('E:\Kuliah\Road to S.T\Semester
8\TA\JST\TAdicky.xlsx', W1, 'W1')
xlswrite('E:\Kuliah\Road to S.T\Semester
8\TA\JST\TAdicky.xlsx', W2, 'W2')

% Tahap Validasi
disp('-----')
disp(' VALIDATION IN PROGRESS ')
disp('-----')

```

```

[n_rows,n_col] = size(uv1);
data_uji = zeros(n_rows-1,sum(hist));

for i = 1:hist(1),
    data_uji(:,i) = [zeros(hist(1)-
i,1);uv1(2:n_rows-hist(1)+i)];
end

for j = 1:hist(2),
    data_uji(:,sum(hist(1))+j) =
[zeros(hist(2)-j,1);uv2(2:n_rows-hist(2)+j)];
end

for k = 1:hist(3),
    data_uji(:,sum(hist(1:2))+k) =
[zeros(hist(3)-k,1);uv3(2:n_rows-hist(3)+k)];
end
for l = 1:hist(4),
    data_uji(:,sum(hist(1:3))+l) =
[zeros(hist(4)-l,1);uv4(2:n_rows-hist(4)+l)];
end

PHI_uji = data_uji';

Y_uji = zeros(n_rows-1,1);
Y_uji(:,1) = yv1(2:end);

Ys_uji = Y_uji';

[y2_uji]=marq_rev_uji(NetDef,W1,W2,PHI_uji,Ys_uj
i);

% RMSE calculation
for i = 1:l
    RMSE_test(i)=
r_m_s_e(Ys_uji(i,:),y2_uji(i,:));

```

```

end

for i = 1:1
    Ys_test(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(Ys_uji(i,:)+1)/2)+min(ys(i,:));
%Descaling
    Yhat_test(i,:)=((max(ys(i,:))-
min(ys(i,:)))*(y2_uji(i,:)+1)/2)+min(ys(i,:));
%Descaling
    %m = Yhat_test(i,:);
    RMSE_test_f(i)=
r_m_s_e(Ys_test(i,:),Yhat_test(i,:));

    %save Yhat_test(i,:)
end

%xlswrite('D:\DOKUMENKU\ITS\TAHUN KE-4\TA
DAPAT A\PROGRESS\Taku.xls', Yhat_test(i,:),
'Yhat_test_transpose')

%xlswrite('D:\DOKUMENKU\ITS\TAHUN KE-4\TA DAPAT
A\PROGRESS\Taku.xls', Yhat_test, 'Yhat_test')
%xlswrite('D:\DOKUMENKU\ITS\TAHUN KE-4\TA DAPAT
A\PROGRESS\Taku.xls', totalerror(i,:),
'totalerror')
%xlswrite('D:\DOKUMENKU\ITS\TAHUN KE-4\TA DAPAT
A\PROGRESS\Taku.xls', RMSE_test_f ,
'RMSE_test_f')

%Drawing
for i = 1
    figure(i+3)
    plot(Ys_test(i,:), 'k', 'LineWidth', 2);
    hold on
    plot(Yhat_test(i,:), 'r.', 'LineWidth', 1);
    grid
    title('Network Validation ');
    legend('Solid : Actual', 'Dot : Predicted
', 'Location', 'Best');

```

```

ylabel('Validasi Clinker');
xlabel('Validation');

%=====
=====

    RMSE_iterasi (i,:)= [RMSE_train RMSE_test
RMSE_train_f RMSE_test_f ];
    Data_RMSE = [Data_RMSE; RMSE_iterasi];

end;

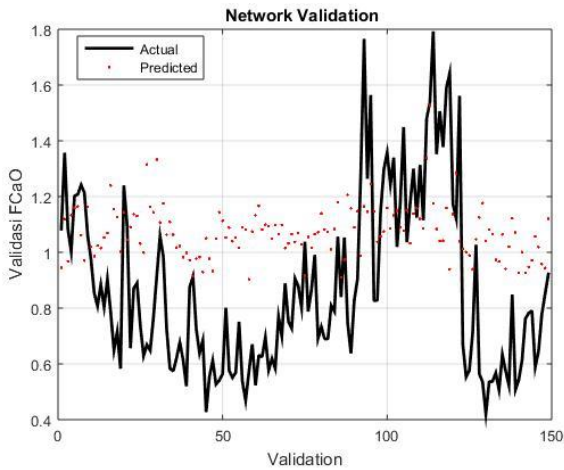
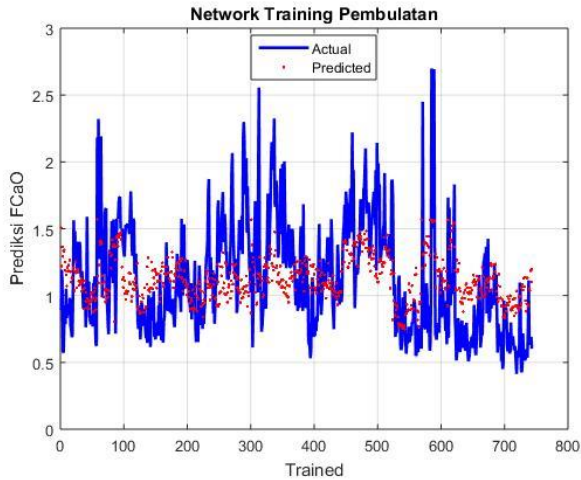
    xlswrite('E:\Kuliah\Road to S.T\Semester
8\TA\JST\TAdicky.xlsx' , RMSE_iterasi, 'RMSE
iterasi 2')
    xlswrite('E:\Kuliah\Road to S.T\Semester
8\TA\JST\TAdicky.xlsx' , Data_RMSE, 'Data RMSE')
    xlswrite('E:\Kuliah\Road to S.T\Semester
8\TA\JST\TAdicky.xlsx' , Yhat_test',
'Yhat_test_transpose')

end;
% save E1101_RMSEfind RMSE_train RMSE_test
RMSE_train_f RMSE_test_f

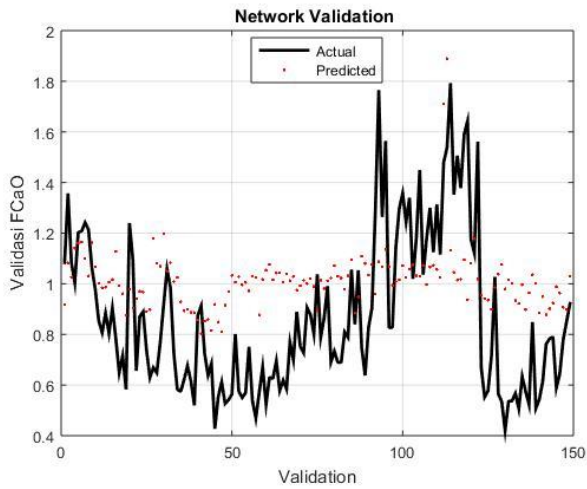
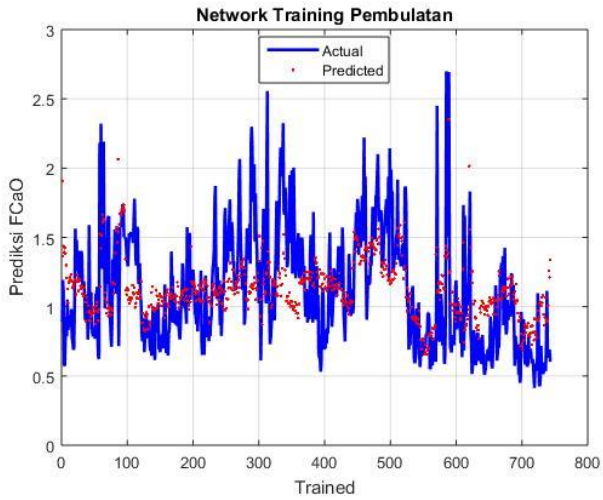
```

Lampiran 2. Grafik Pelatihan dan Validasi Data Prediksi Persentase FCaO dengan Metode JST

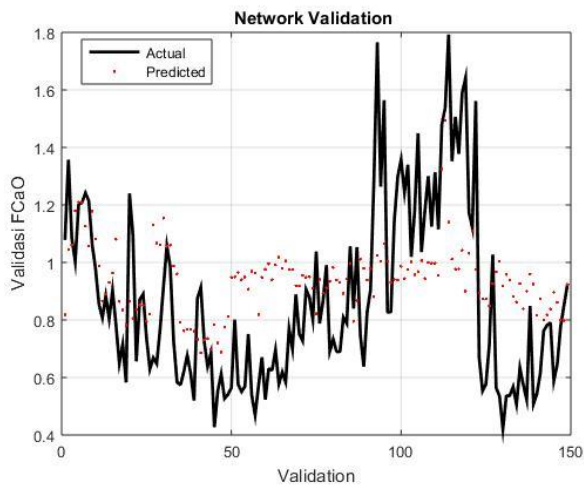
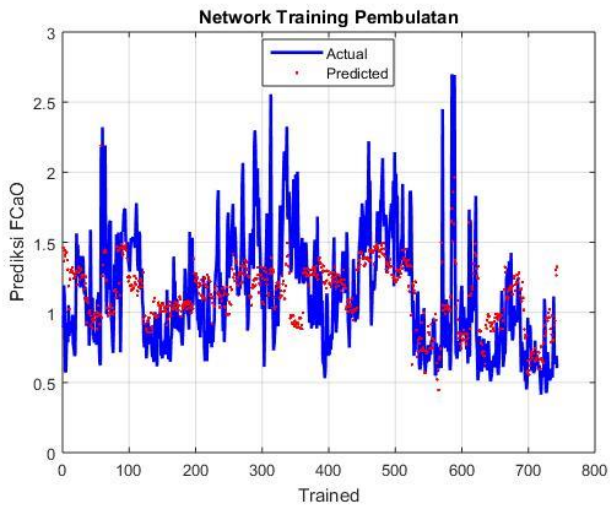
- 1 Hidden Node



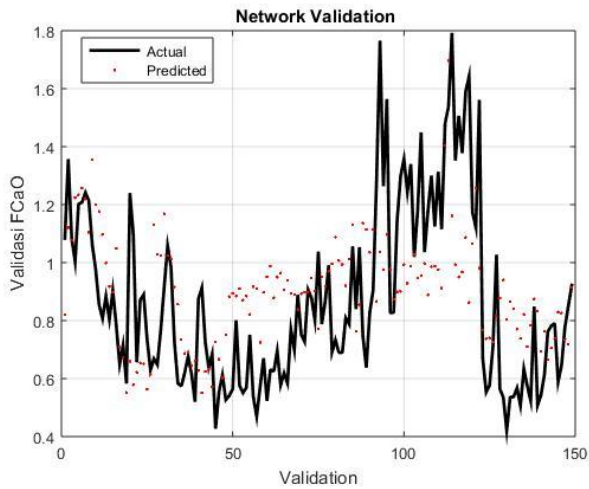
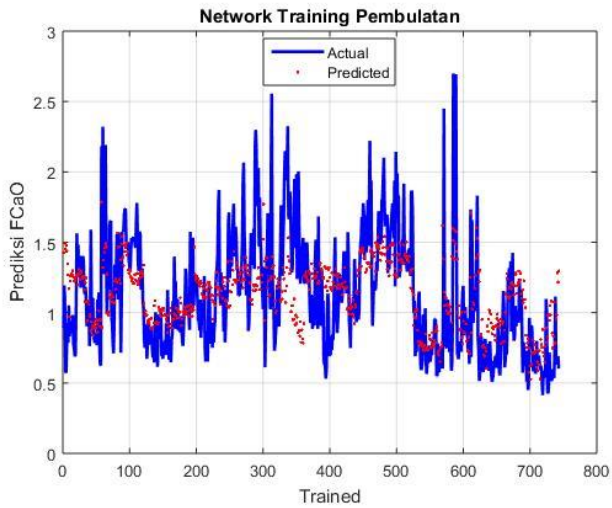
- 2 Hidden Node



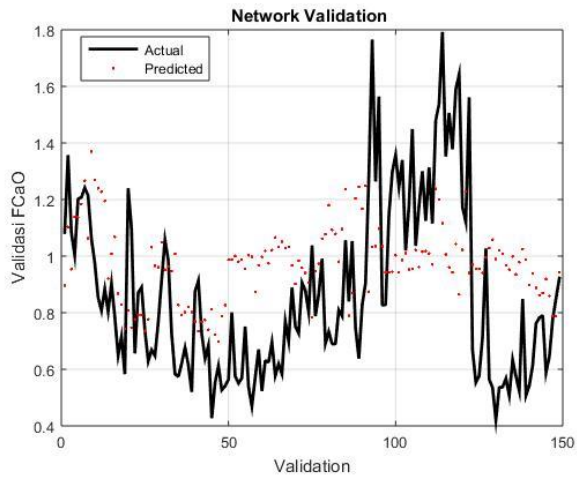
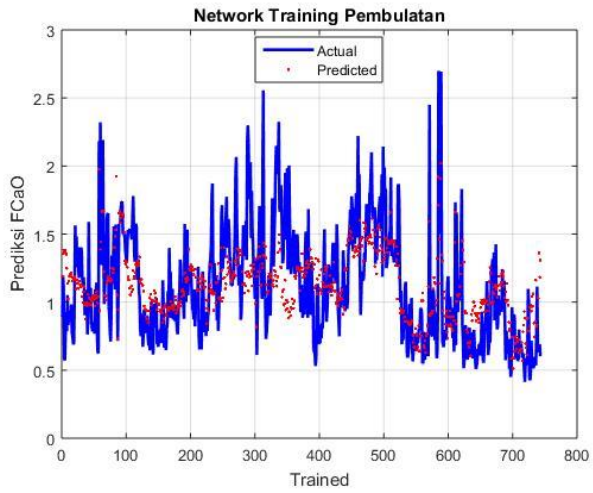
- 3 Hidden Node



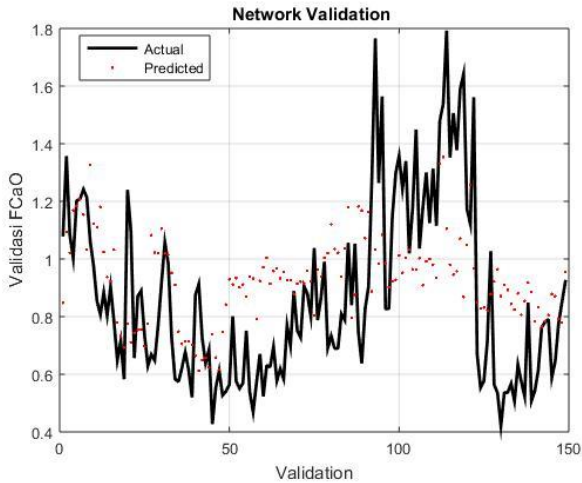
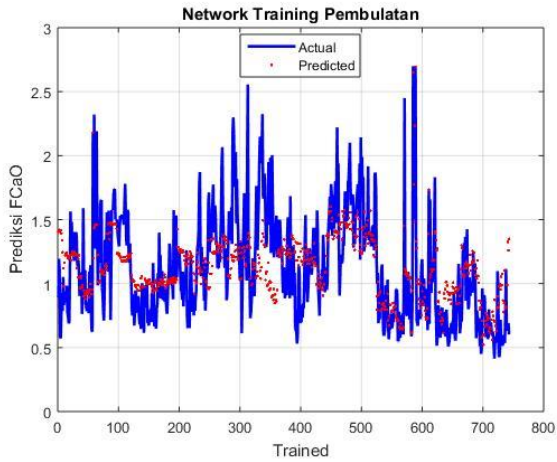
- 4 Hidden Node



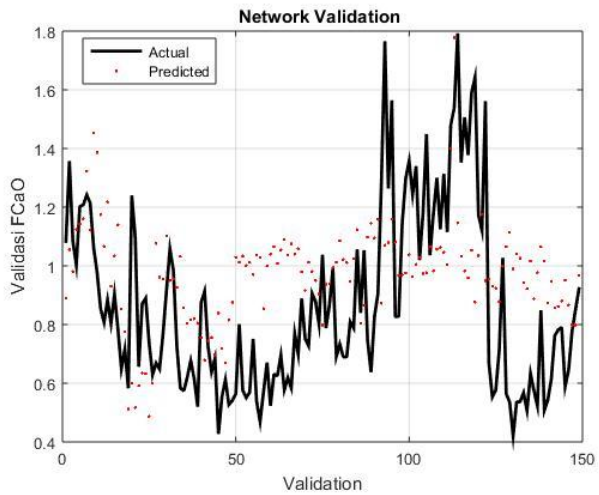
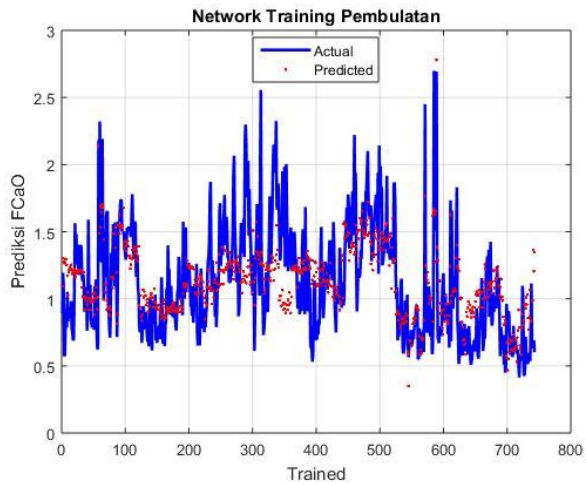
- 5 Hidden Node



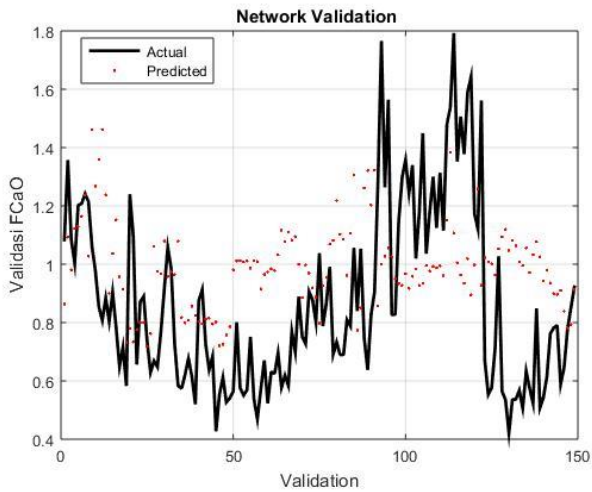
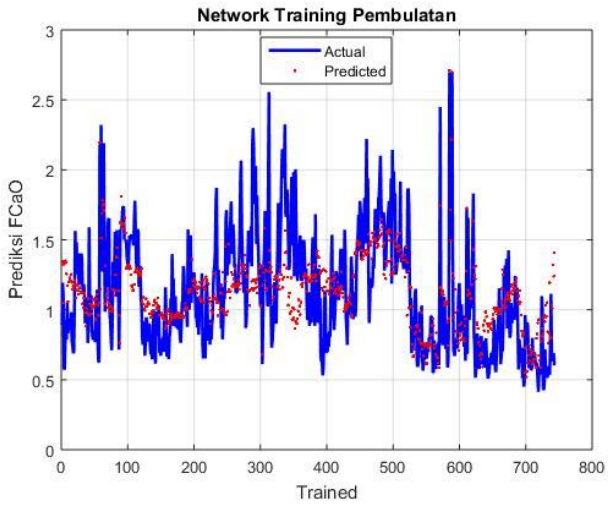
- 7 Hidden Node



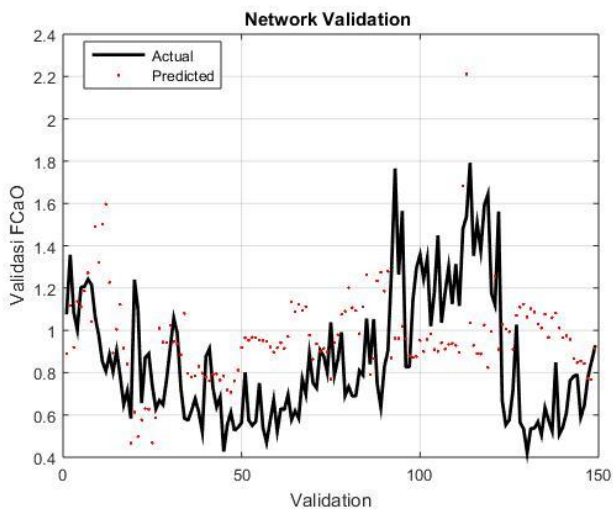
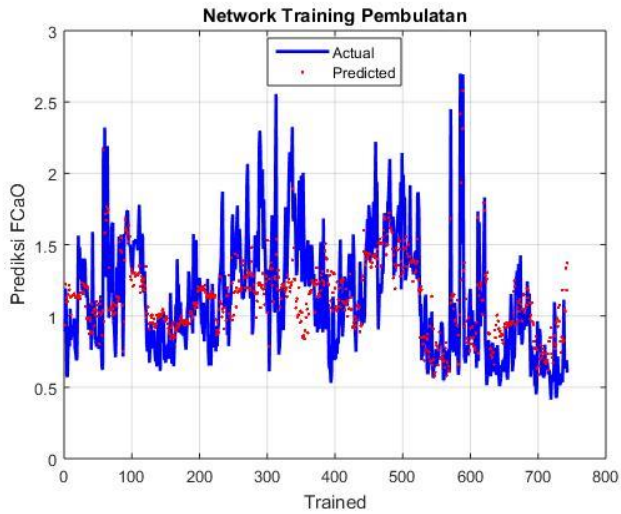
- 8 Hidden Node



- 9 Hidden Node



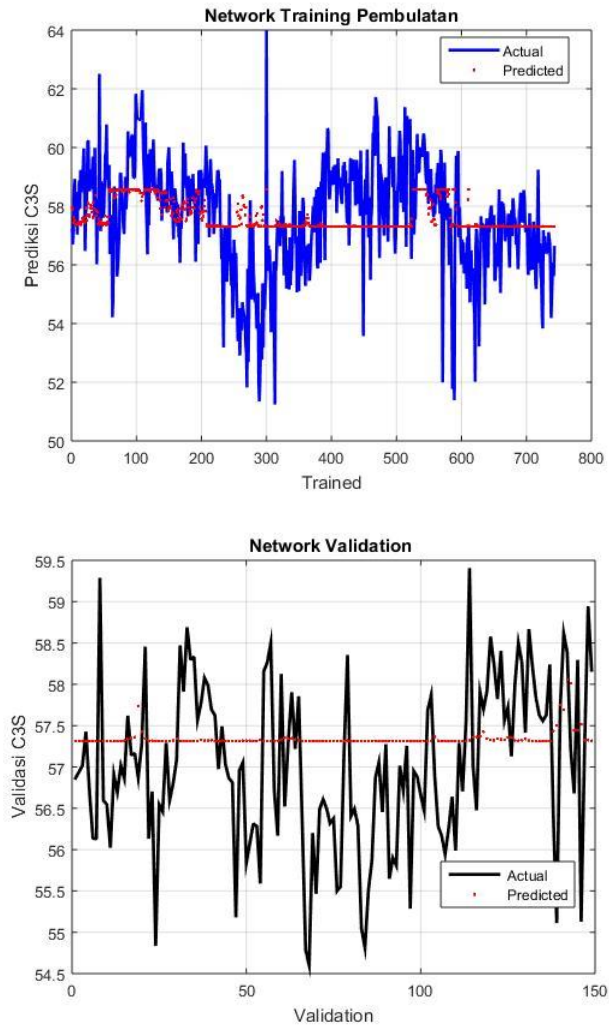
- 10 *Hidden Node*



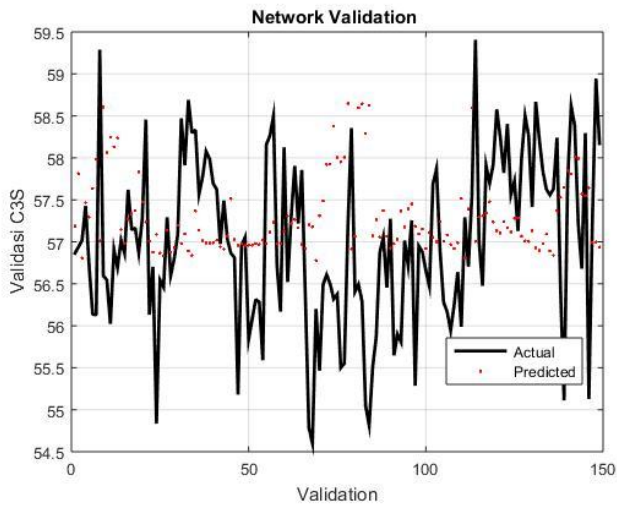
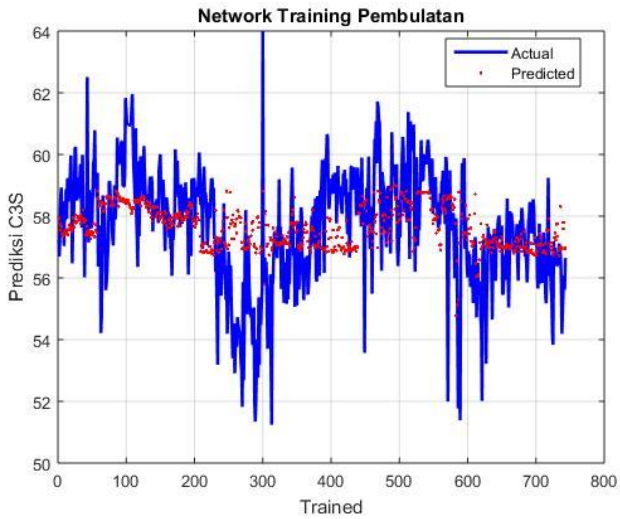
“Halaman ini sengaja dikosongkan”

Lampiran 4. Grafik Pelatihan dan Validasi Data Prediksi Persentase C3S dengan Metode JST

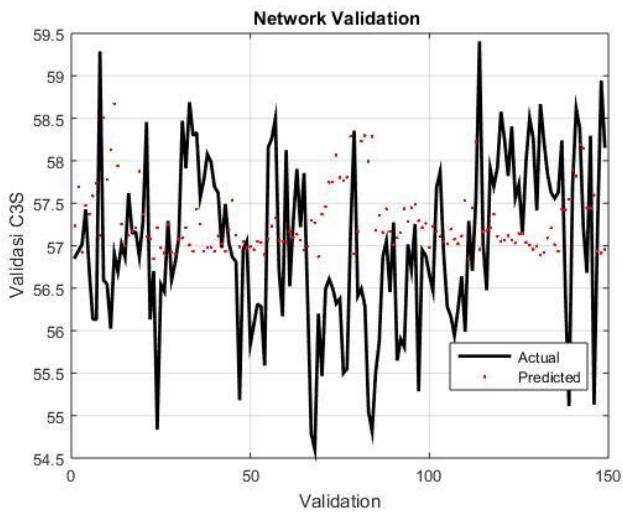
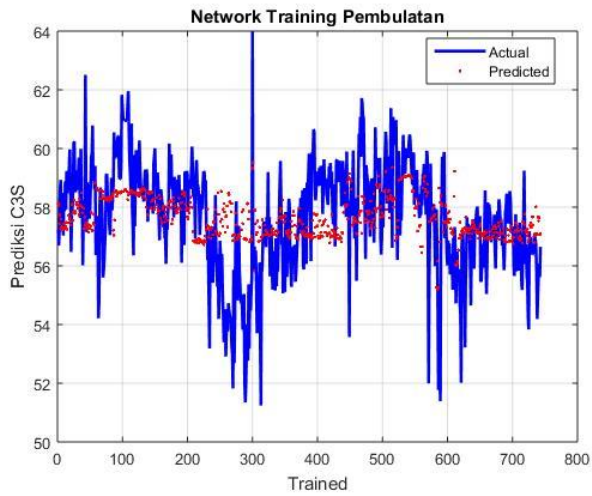
- 1 Hidden Node



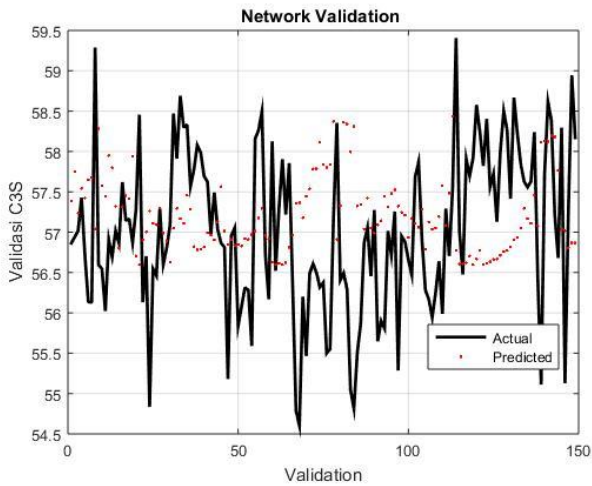
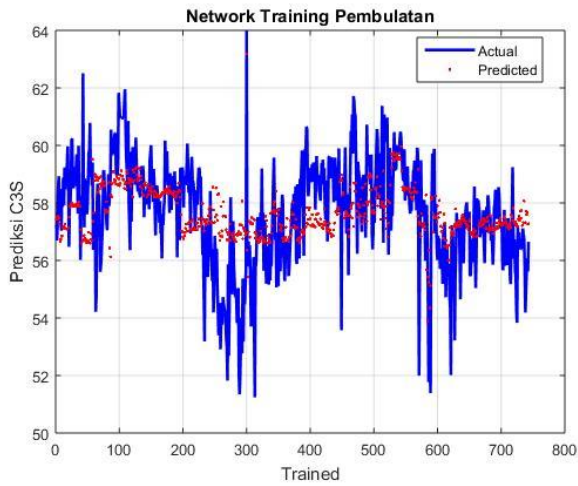
- 2 Hidden Node



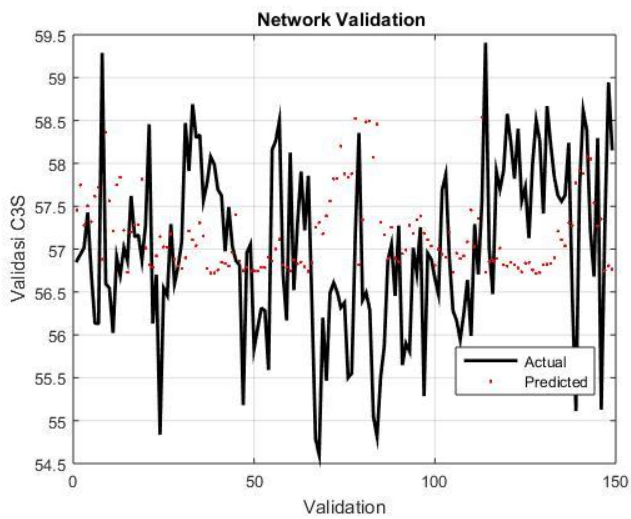
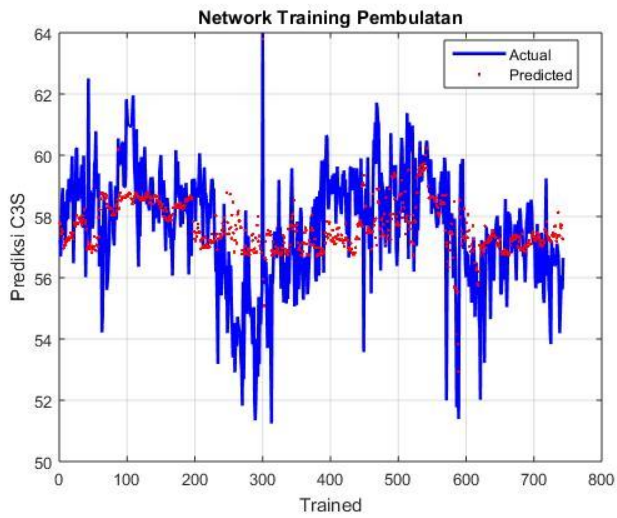
- 3 Hidden Node



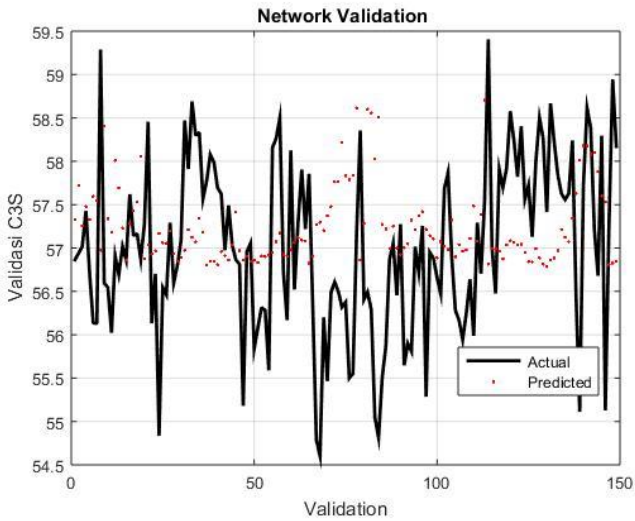
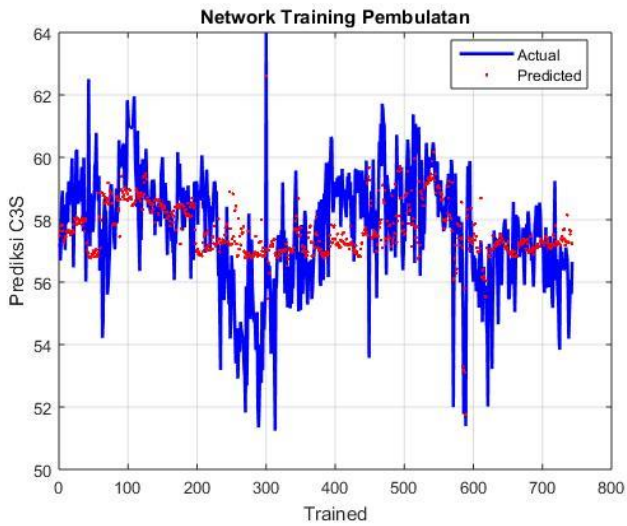
- 5 Hidden Node



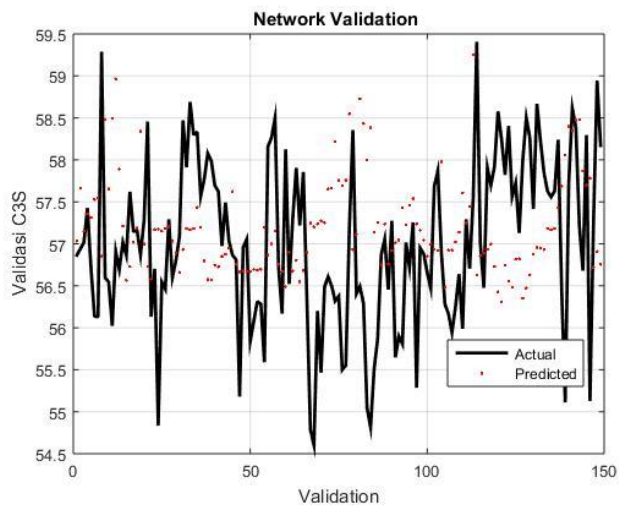
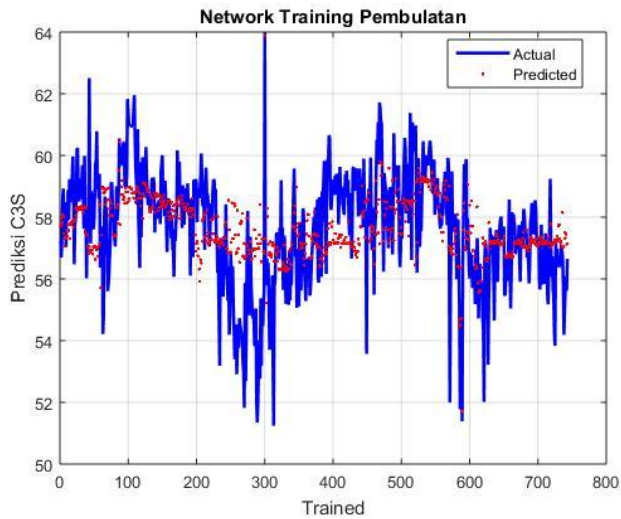
- 6 Hidden Node



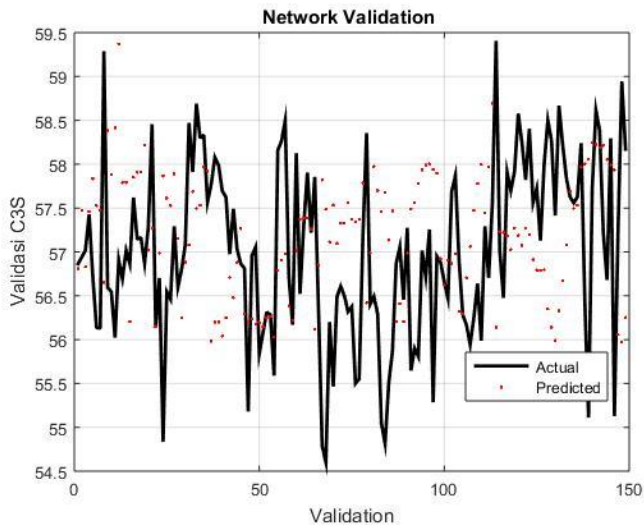
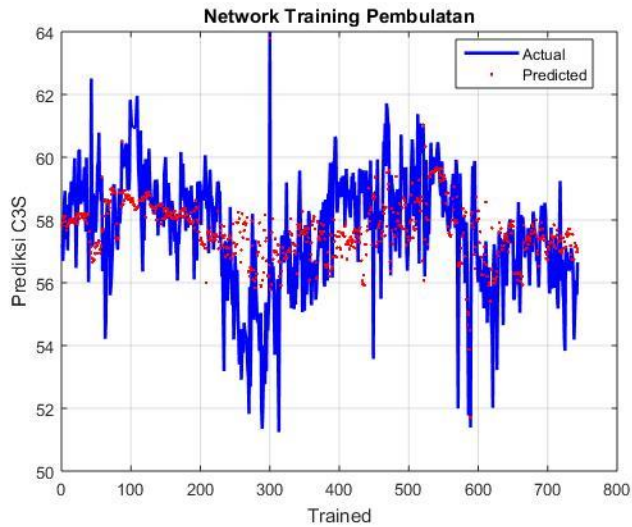
- 7 Hidden Node



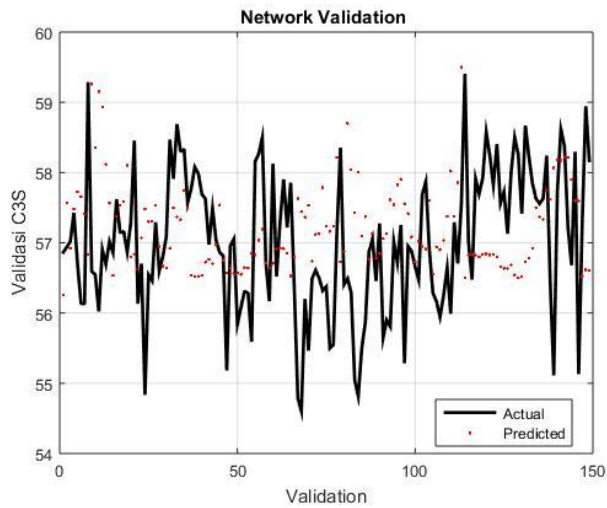
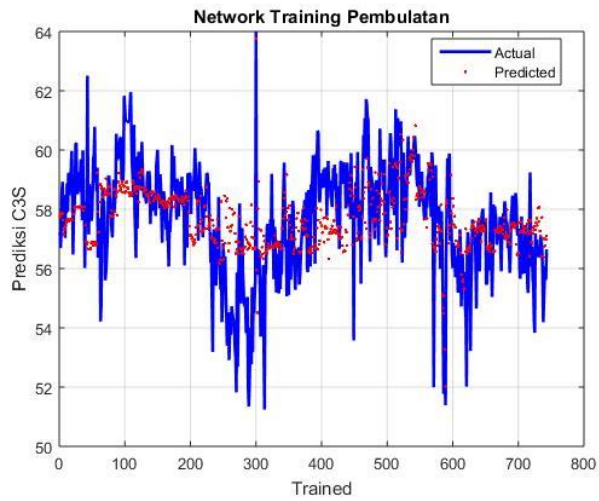
- 8 Hidden Node



- 9 Hidden Node



- 10 Hidden Node



“Halaman ini sengaja dikosongkan”

Lampiran 4. Koding Optimisasi Jaringan Syaraf Tiruan dan Genetic Algorithm

```
clear all
close all
clc
%GENETIC ALGORITHM%
%Pembangkitan Populasi dan Parameter
% load ('prop.mat')

Dimension=7;
UB = [ 0.8 0.7 -0.2 0.2 1.4 ...
      -47 25]; % Upper Bounds
LB = [ 0.6 0.5 -0.4 0.09 1.2 ...
      -48 24 ]; % Lower Bounds
% 0,736196159    0,681574652 -0,293820037
0,113468074 1,377133024

%-47,66845062    25,58924259

Npop      = 20;           %populasi
Maxit     = 10;           %iterasi
el        = 0.95;         %elatism
Pc        = 0.8;
%probabilitas crossover
Pm        = 0.001;
%probabilitas mutasi
Nbit      = 20;           %jumlah bit
%Constrain

eBangkit   = [];
Individu   = [];
eIndividu  = [];
david      = [];
Dadatfit   = [];
Datfit     = [];
summary    = [];
```

```

eDadatfit    = [];
efitnessmax  = [];
eIndividuMax = [];

Bangkit = round(rand(Npop,Nbit*Dimension));
popsize = size(Bangkit,1);

for k = 1:Dimension
    batas(k) = UB(k)-LB(k);
end

for i =1:Npop
    for j = 1:Dimension
        Desimal(i,j) =
bi2de(Bangkit(i, ((j*Nbit)-(Nbit-
1)): (j*Nbit)), 'left-msb');
        Individu(i,j) =
(Desimal(i,j)*batas(:,j)-
batas(:,j)+LB(:,j)*(2^Nbit-1))/(2^Nbit-1);
    end
end

Datfit = [];
variabel = [];

for i = 1:size(Individu,1)
    fitness = Clinker(Individu(i,:));
    Datfit = [Datfit;fitness];
    [fitemax,nmax]=max(Datfit);
end

Dadatfit = [];
for generasi=1:Maxit
    disp('GA processing')
    clear command windows
    clear command history
    clear memory

    if generasi > 1

```

```

        sort_fit =
sortrows(sort,Nbit*Dimension+1);
        Individul = sort_fit(round((1-
el)*Npop+1):Npop,:);
        remain =
sort_fit(round(el*Npop)+1:Npop,:);

X = Individul;
M = size(X,1);

sumfitness = sum(Datfit);
for i=1:M
    Prob(i) = Datfit(i)/sumfitness;
end
for i=2:M
    Prob(i) = Prob(i)+Prob(i-1);
end
for i=1:M
    n=rand;
    k=1;
    for j=1:M-1
        if (n>Prob(j))
            k=j+1;
        end
    end
    Xparents(i,:) = X(k,:);
end

%Crossover
[M,d] = size(Xparents);
Xcrossed = Xparents;
for i=1:2:M-1
    c=rand;
    if (c<=Pc)
        p=ceil((d-1)*rand);
        Xcrossed(i,:) = [Xparents(i,1:p)
Xparents(i+1,p+1:d)];
        Xcrossed(i+1,:) =
[Xparents(i+1,1:p) Xparents(i,p+1:d)];
    end

```

```

end
if (M/2~=floor(M/2))
    c=rand;
    if (c<=Pc)
        p=ceil((d-1)*rand);
        str=ceil((M-1)*rand);
        Xcrossed(M,:) = [Xparents(M,1:p)
Xparents(str,p+1:d)]; %the first child is chosen
    end
end

%Mutasi
[M,d] = size(Xcrossed);
Xnew=Xcrossed;
for i=1:M
    for j=1:d
        p=rand;
        if (p<=Pm)
            Xnew(i,j)=1-Xcrossed(i,j);
        end
    end
end

disp('New fitness calculation');

Bangkit =
[Xnew(:,1:Nbit*Dimension);remain(:,1:Nbit*Dimens
ion)];
end
eBangkit = [eBangkit; Bangkit];

for i =1:Npop
    for j = 1:Dimension;
        Desimal(i,j) =
bi2de(Bangkit(i,((j*Nbit)-(Nbit-
1)):(j*Nbit)), 'left-msb');
        Individu(i,j) =
(Desimal(i,j)*batas(:,j)-
batas(:,j)+LB(:,j)*(2^Nbit-1))/(2^Nbit-1);
    end

```

```

end

Datfit = [];
for i = 1:Npop
    fitness = Clinker(Individu(i,:));
    Datfit = [Datfit;fitness];
    [fitemax,nmax] = max(Datfit);
end

Dadatfit = Datfit;
eDadatfit = [eDadatfit;Dadatfit];
eIndividu = [eIndividu;Individu];
[fitnessmax,nmax] = max(eDadatfit);
efitnessmax = [efitnessmax;fitnessmax];
BangkitMax = eBangkit(nmax,:);
IndividuMax = eIndividu(nmax,:);
eIndividuMax = [eIndividuMax;IndividuMax];
BangkitMaxlast = BangkitMax;
schedmax = BangkitMax;
sort = [Bangkit Dadatfit];
summary = [summary; sort];
david = [david; Dadatfit];

clc
max_variable_design=IndividuMax(1,:)
max_objective_function=fitness(1,:)
figure(gcf)
title('Grafik Nilai Maksimum GA','color','b')
xlabel('Jumlah Iterasi')
ylabel('Nilai Fungsi Obyektif')
hold on
plot(efitnessmax, 'DisplayName', 'efitnessmax',
'YDataSource', 'efitnessmax');
hold on

end
save ('GA.mat')

```

“Halaman ini sengaja dikosongkan”

Lampiran 5. Koding fungsi objektif untuk optimisasi dengan menggunakan 10 *Hidden Node*

```
function NilaiRMSE=Clinker(x)
validasi = xlsread('E:\Kuliah\Road to
S.T\Semester
8\TA\JST\JSTdanGA.xlsx','KilnNorm','E1:E893');
%panggil data
variabelinput = xlsread('E:\Kuliah\Road to
S.T\Semester
8\TA\JST\JSTdanGA.xlsx','KilnNorm','A1:G893');
%TERMASUK BIAS PADA HIDDEN LAYER
```

```
%Fungsi Obyektif
```

```
    t1=x(1);
    t2=x(2);
    t3=x(3);
    t4=x(4);
    t5=x(5);

    t6=x(6);
    t7=x(7);
    t8=x(8);
    t9=x(9);
    t10=x(10);

    t11=x(11);
    t12=x(12);
    t13=x(13);
    t14=x(14);
    t15=x(15);

    t16=x(16);
    t17=x(17);
    t18=x(18);
    t19=x(19);
    t20=x(20);

    t21=x(21);
    t22=x(22);
```

t23=x (23) ;
t24=x (24) ;
t25=x (25) ;

t26=x (26) ;
t27=x (27) ;
t28=x (28) ;
t29=x (29) ;
t30=x (30) ;

t31=x (31) ;
t32=x (32) ;
t33=x (33) ;
t34=x (34) ;
t35=x (35) ;

t36=x (36) ;
t37=x (37) ;
t38=x (38) ;
t39=x (39) ;
t40=x (40) ;

t41=x (41) ;
t42=x (42) ;
t43=x (43) ;
t44=x (44) ;
t45=x (45) ;

t46=x (46) ;
t47=x (47) ;
t48=x (48) ;
t49=x (49) ;
t50=x (50) ;

t51=x (51) ;
t52=x (52) ;
t53=x (53) ;
t54=x (54) ;
t55=x (55) ;
t56=x (56) ;

```
t57=x(57);  
t58=x(58);  
t59=x(59);  
t60=x(60);  
t61=x(61);
```

```
a = variabelinput(1:893,1)*t1;  
b = variabelinput(1:893,2)*t2;  
c = variabelinput(1:893,3)*t3;  
d = variabelinput(1:893,4)*t4;  
e = ones(893,1)*t5;
```

```
hn1 = [a b c d e];
```

```
a2 = variabelinput(1:893,1)*t6;  
b2 = variabelinput(1:893,2)*t7;  
c2 = variabelinput(1:893,3)*t8;  
d2 = variabelinput(1:893,4)*t9;  
e2 = ones(893,1)*t10;
```

```
hn2 = [a2 b2 c2 d2 e2];
```

```
a3 = variabelinput(1:893,1)*t11;  
b3 = variabelinput(1:893,2)*t12;  
c3 = variabelinput(1:893,3)*t13;  
d3 = variabelinput(1:893,4)*t14;  
e3 = ones(893,1)*t15;
```

```
hn3 = [a3 b3 c3 d3 e3];
```

```
a4 = variabelinput(1:893,1)*t16;  
b4 = variabelinput(1:893,2)*t17;  
c4 = variabelinput(1:893,3)*t18;  
d4 = variabelinput(1:893,4)*t19;  
e4 = ones(893,1)*t20;
```

```
hn4 = [a4 b4 c4 d4 e4];
```

```
a5 = variabelinput(1:893,1)*t21;
```

```
b5 = variabelinput(1:893,2)*t22;  
c5 = variabelinput(1:893,3)*t23;  
d5 = variabelinput(1:893,4)*t24;  
e5 = ones(893,1)*t25;
```

```
hn5 = [a5 b5 c5 d5 e5];
```

```
a6 = variabelinput(1:893,1)*t26;  
b6 = variabelinput(1:893,2)*t27;  
c6 = variabelinput(1:893,3)*t28;  
d6 = variabelinput(1:893,4)*t29;  
e6 = ones(893,1)*t30;
```

```
hn6 = [a6 b6 c6 d6 e6];
```

```
a7 = variabelinput(1:893,1)*t31;  
b7 = variabelinput(1:893,2)*t32;  
c7 = variabelinput(1:893,3)*t33;  
d7 = variabelinput(1:893,4)*t34;  
e7 = ones(893,1)*t35;
```

```
hn7 = [a7 b7 c7 d7 e7];
```

```
a8 = variabelinput(1:893,1)*t36;  
b8 = variabelinput(1:893,2)*t37;  
c8 = variabelinput(1:893,3)*t38;  
d8 = variabelinput(1:893,4)*t39;  
e8 = ones(893,1)*t40;
```

```
hn8 = [a8 b8 c8 d8 e8];
```

```
a9 = variabelinput(1:893,1)*t41;  
b9 = variabelinput(1:893,2)*t42;  
c9 = variabelinput(1:893,3)*t43;  
d9 = variabelinput(1:893,4)*t44;  
e9 = ones(893,1)*t45;
```

```
hn9 = [a9 b9 c9 d9 e9];
```

```
a10 = variabelinput(1:893,1)*t46;  
b10 = variabelinput(1:893,2)*t47;  
c10 = variabelinput(1:893,3)*t48;  
d10 = variabelinput(1:893,4)*t49;  
e10 = ones(893,1)*t50;
```

```
hn10 = [a10 b10 c10 d10 e10];
```

```
for i = 1:893;  
    nilaihn1(i,1) = sum(hn1(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn2(i,1) = sum(hn2(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn3(i,1) = sum(hn3(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn4(i,1) = sum(hn4(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn5(i,1) = sum(hn5(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn6(i,1) = sum(hn6(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn7(i,1) = sum(hn7(i,1:5));  
end;
```

```
for i = 1:893;  
    nilaihn8(i,1) = sum(hn8(i,1:5));  
end;
```

```

for i = 1:893;
    nilaihn9(i,1) = sum(hn9(i,1:5));
end;

for i = 1:893;
    nilaihn10(i,1) = sum(hn10(i,1:5));
end;

%FUNGSI AKTIVASI HN

fhn1 = tanh(nilaihn1)*t51;
fhn2 = tanh(nilaihn2)*t52;
fhn3 = tanh(nilaihn3)*t53;
fhn4 = tanh(nilaihn4)*t54;
fhn5 = tanh(nilaihn5)*t55;
fhn6 = tanh(nilaihn6)*t56;
fhn7 = tanh(nilaihn7)*t57;
fhn8 = tanh(nilaihn8)*t58;
fhn9 = tanh(nilaihn9)*t59;
fhn10 = tanh(nilaihn9)*t60;
fhn11 = ones(893,1)*t61;
matrix=[fhn1,fhn2,fhn3,fhn4,fhn5,fhn6,fhn7,fhn8,
fhn9,fhn10,fhn11];

for i = 1:893;
    nilaion(i,1)= sum(matrix(i,1:11));

end;

%LAYER OUTPUT FUNGSI AKTIVASI
yprediksi = tanh(nilaion);

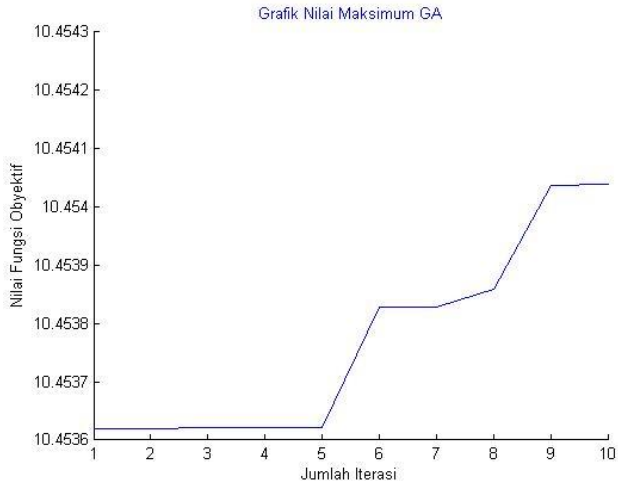
%objective function
pengurangan = yprediksi - validasi;
pengurangankuadrat = pengurangan.^2;

%Sehingga
NilaiRMSE = sqrt(sum(pengurangankuadrat))/893;

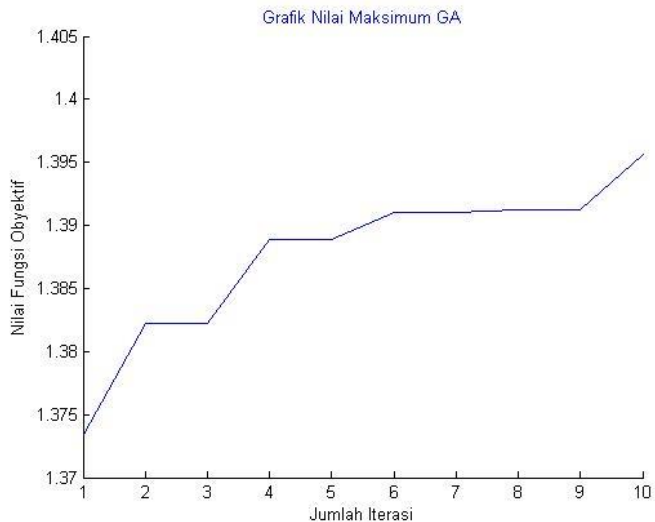
```

Lampiran 6. Grafik Performansi JST-GA pada Prediksi Presentase FCaO

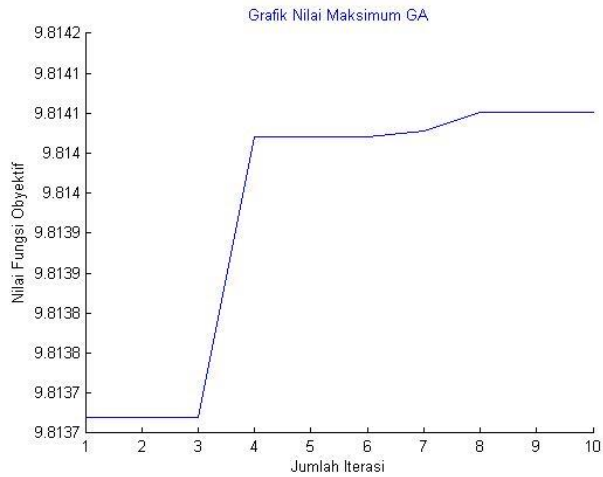
- 1 Hidden Node



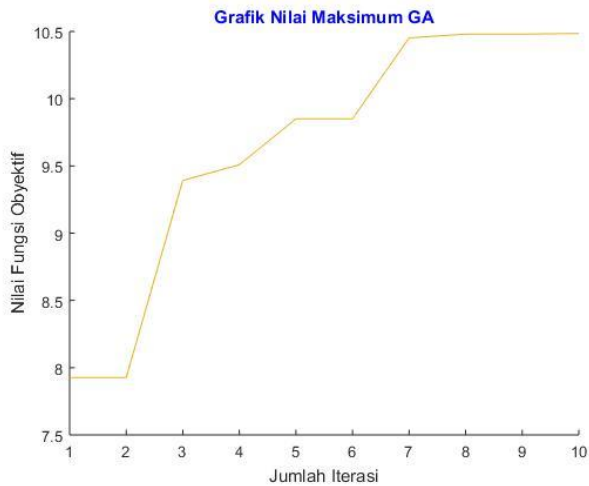
- 3 Hidden Node



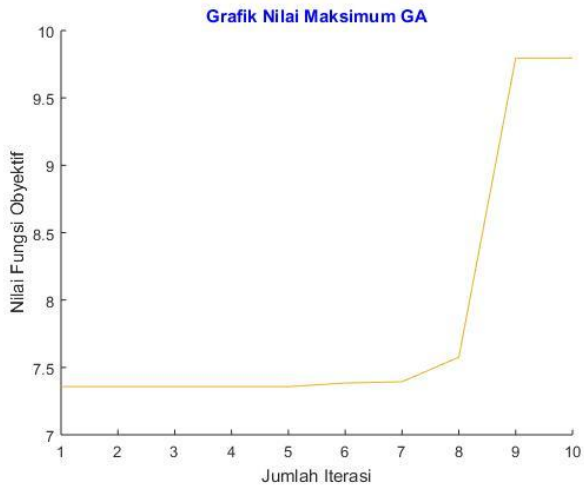
- 4 Hidden Node



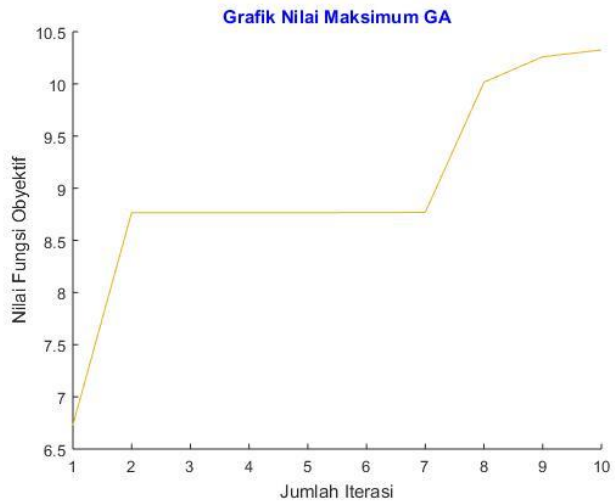
- 5 Hidden Node



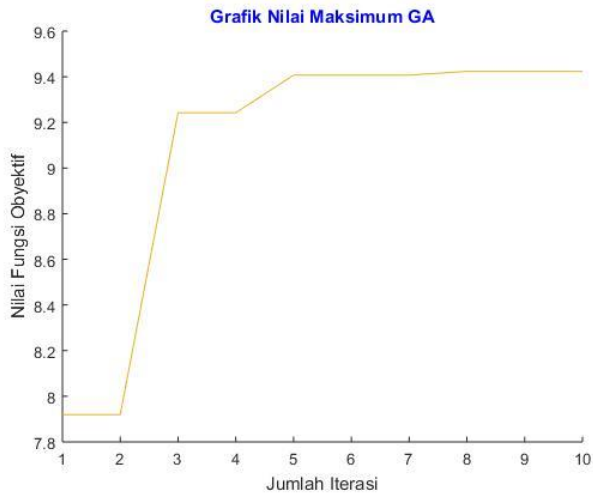
- 6 Hidden Node



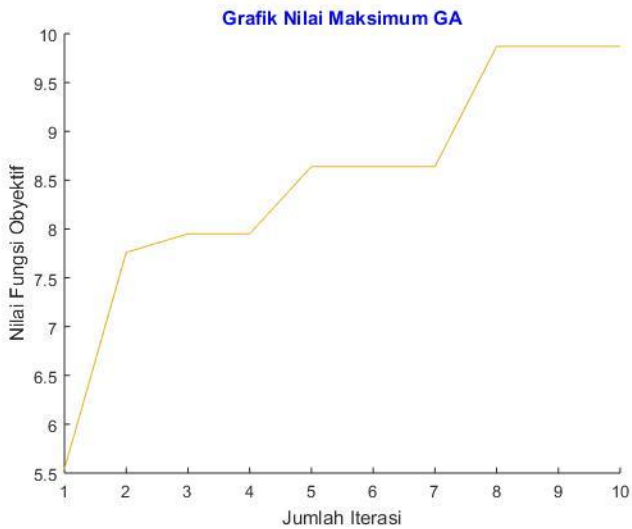
- 7 Hidden Node



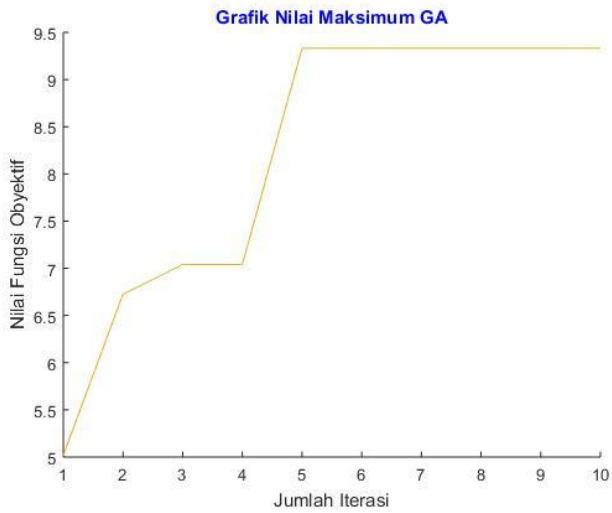
- 8 Hidden Node



- 9 Hidden Node



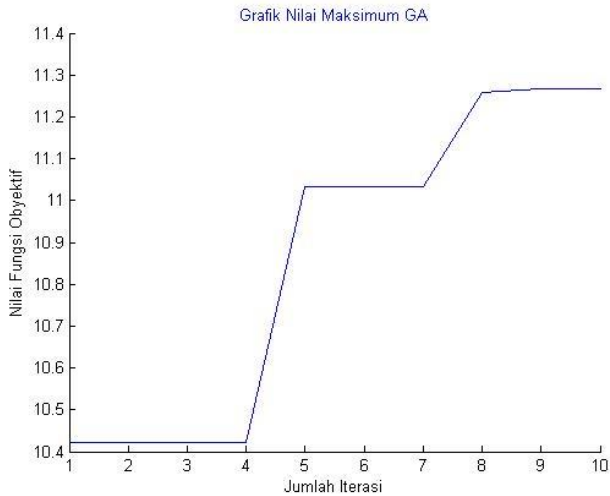
- 10 *Hidden Node*



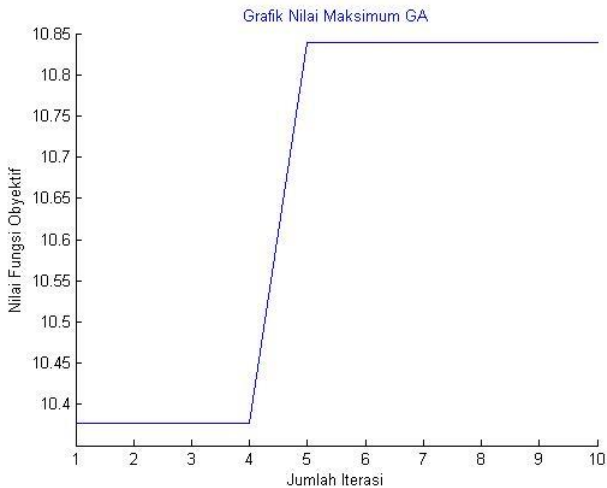
“Halaman ini sengaja dikosongkan”

Lampiran 7. Grafik Performansi JST-GA pada Prediksi Presentase FCaO

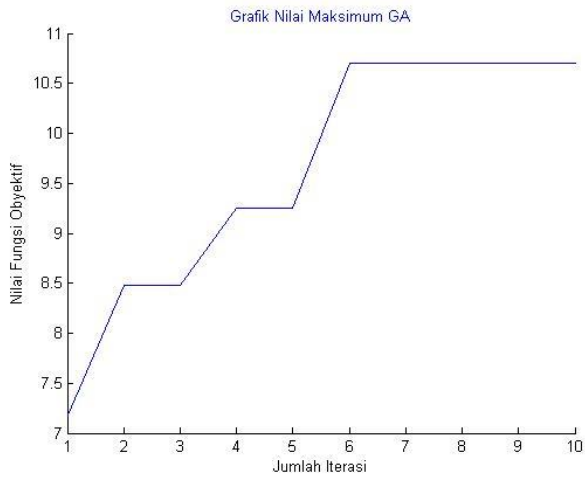
- 2 Hidden Node



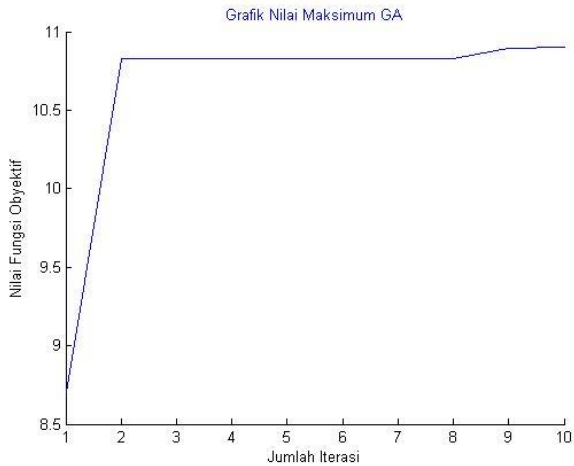
- 3 Hidden Node



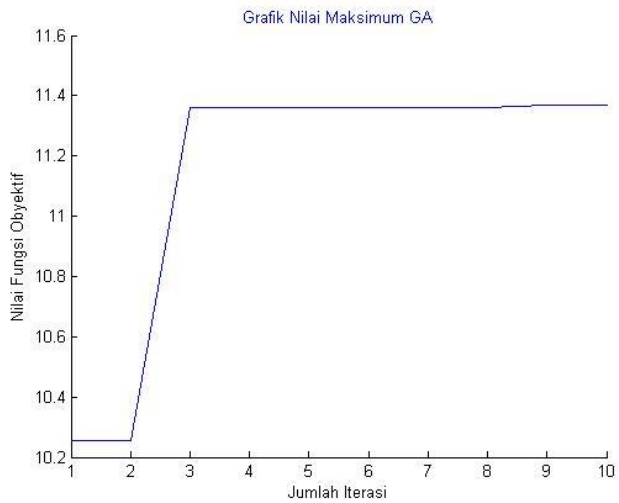
- 4 *Hidden Node*



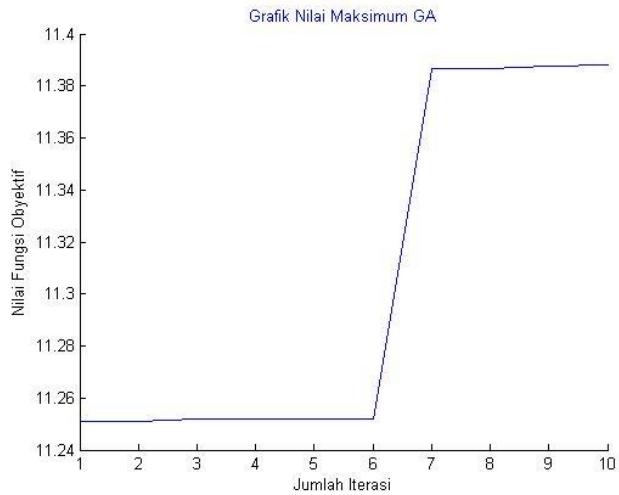
- 5 *Hidden Node*



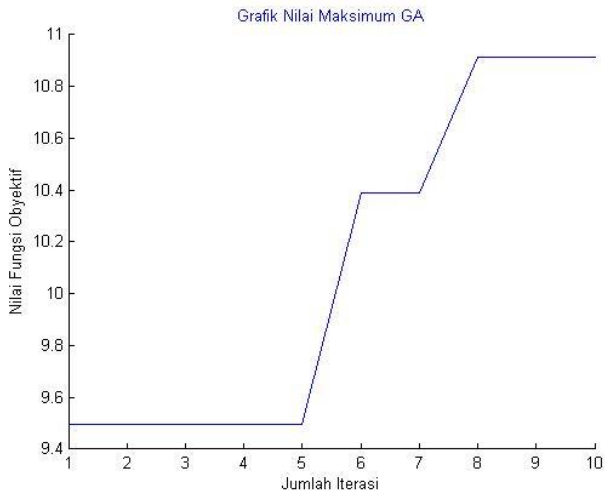
- *6 Hidden Node*



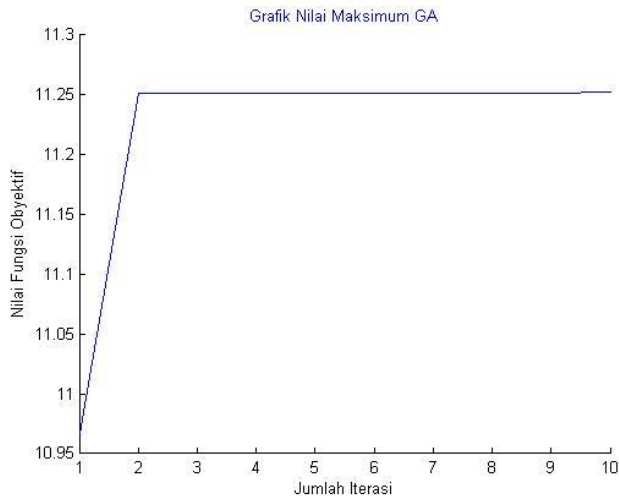
- *7 Hidden Node*



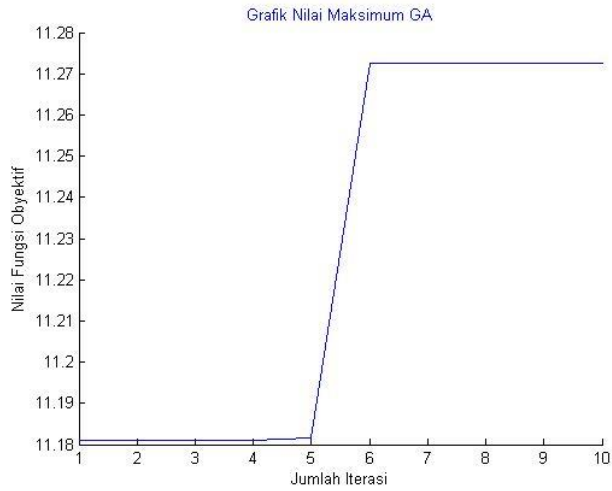
- *8 Hidden Node*



- *9 Hidden Node*



- 10 *Hidden Node*



“Halaman ini sengaja dikosongkan”

BIODATA PENULIS



Penulis mempunyai nama lengkap **Dicky Dwi Arifiansyah**, lahir di Gresik pada tanggal 16 Agustus 1995, merupakan anak ke dua dari tiga bersaudara. Penulis menempuh pendidikan sekolah dasar di SD Negeri Sukowati, pendidikan menengah di SMP Negeri 1 Bungah, dan melanjutkan pendidikan menengah atas di SMA Negeri 1 Manyar yang kemudian dilanjutkan ke ITS Jurusan Teknik Fisika pada tahun 2013 dengan NRP 2413100118. Hingga pada tahun 2017 penulis akhirnya melaksanakan Tugas Akhir sebagai syarat untuk menyelesaikan studi S1 di Jurusan Teknik Fisika dengan judul “**Optimisasi Pembakaran di *Rotary Kiln* PT Semen Gresik Pabrik Tuban Dengan Model Jaringan Syaraf Tiruan dan Metode *Genetic Algorithm***”. Bagi pembaca yang memiliki saran, kritik, atau ingin berdiskusi lebih lanjut tentang Tugas Akhir ini, maka bisa menghubungi penulis melalui dickydwiarif@gmail.com.